

**Министерство образования и науки РФ**  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
Сибирский федеральный университет  
**Лесосибирский педагогический институт –**  
филиал федерального государственного автономного образовательного  
учреждения высшего профессионального образования  
**Сибирский федеральный университет**

# **ПРОГРАММИРОВАНИЕ**

Комплекс лабораторных работ

Лесосибирск 2011

Программирование. Комплекс лабораторных работ:/ Составитель Е.В.Киргизова. – Лесосибирск: ЛПИ-филиал СФУ, 2011. – 44с.

В комплексе лабораторных работ приведены краткие теоретические сведения по основам программирования в среде DELPHI, а также по языку программирования Object Pascal.

## МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ СТУДЕНТОВ ПО ДИСЦИПЛИНЕ «ПРОГРАММИРОВАНИЕ»

Методические указания составлены с учетом современных требований к подготовке будущих учителей информатики.

Данные методические указания позволяют получить основные практические навыки работы со средой объектно-ориентированного программирования Borland Delphi, ознакомиться с ее основными возможностями

Методические рекомендации для студентов содержат:

- планы семинарских занятий с указанием примерного списка аудиторных заданий;
- планы лабораторных работ с указанием примерного списка аудиторных и домашних заданий.

В ходе выполнения лабораторных работ студенты изучат приемы и способы обработки:

- символьных данных;
- числовых данных;
- массивов;
- графических изображений;
- табличных данных.

Так же будут рассмотрены основные способы создания интерфейса программного средства, контекстной справки, помощи и способы вывода данных на печать.

Вопросы, изученные в ходе выполнения данных лабораторных работ, помогут студентам полностью освоить работу в среде визуального и объектно-ориентированного программирования Borland Delphi.

### **Лабораторная работа № 1**

*Среда объектно-ориентированного проектирования DELPHI*

**Цель:** изучить интерфейс и базовые возможности Borland Delphi.

#### **ЗНАКОМСТВО СО СРЕДОЙ DELPHI**

Среда Delphi - это сложный механизм, обеспечивающий высокоэффективную работу программиста. Визуально она реализуется несколькими одновременно раскрытыми на экране окнами. Окна могут перемещаться по экрану, частично или полностью перекрывая друг друга. Каждое окно несет в себе некоторую функциональность, т. е. предназначено для решения определенных задач.

При запуске Delphi вы увидите рабочее окно среды программирования (Смотри рисунок 1). На рисунке показаны окна Delphi версии 7. На нем изображены: главное окно, окно Дерева объектов (Object Tree View), окно Инспектора объектов, окно браузера, окно формы и окно кода программы.

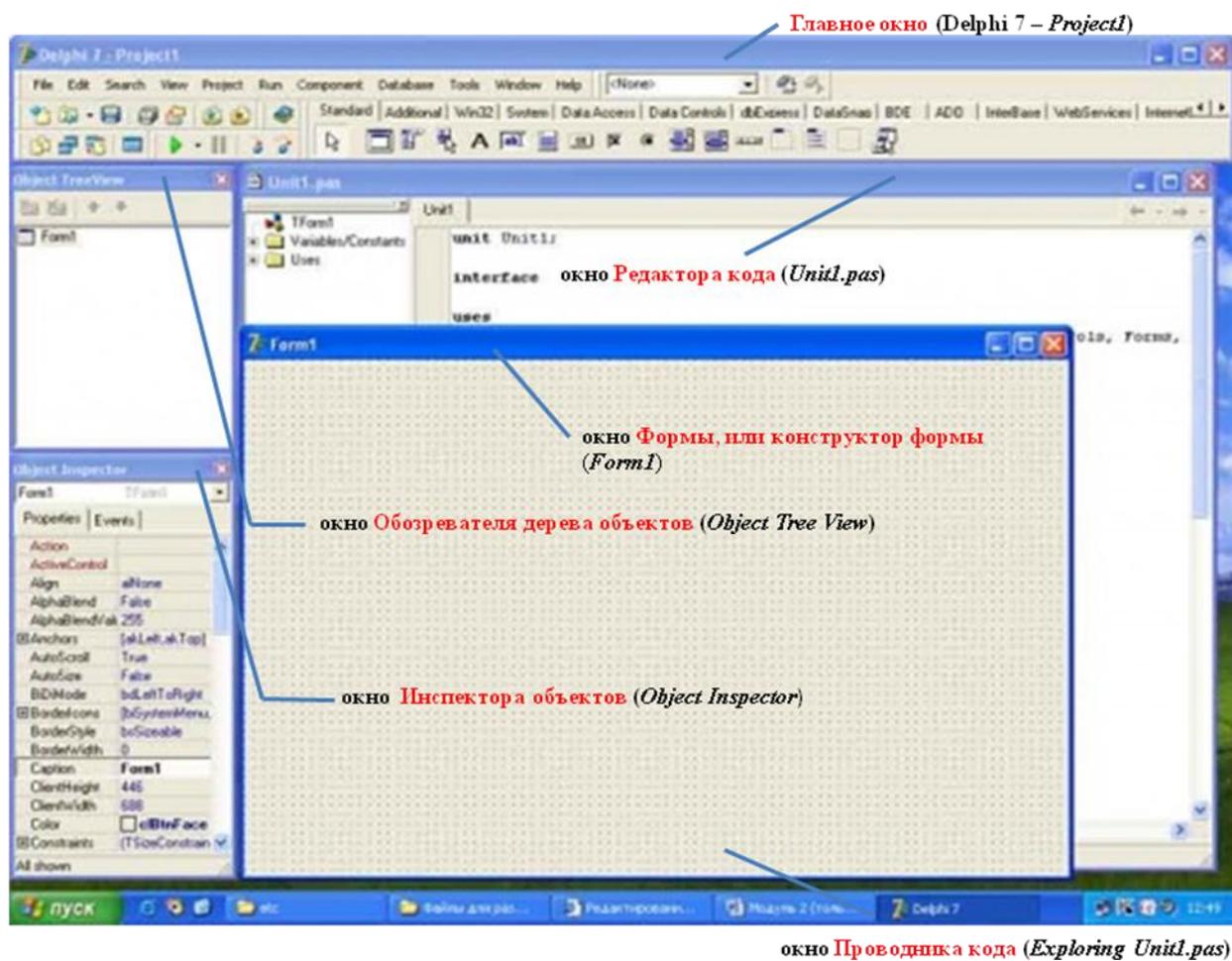


Рисунок 1 - Окна Delphi

Главное окно осуществляет основные функции управления проектом создаваемой программы.

В главном окне располагается главное меню Delphi, набор пиктографических командных кнопок и палитра компонентов.

Пиктографические кнопки открывают быстрый доступ к наиболее важным опциям главного меню. По функциональному признаку они разделены на 6 групп:

- |                                  |                                  |
|----------------------------------|----------------------------------|
| 1. <b>Standard</b> (Стандартная) | 4. <b>Custom</b> (Пользователь)  |
| 2. <b>View</b> (Просмотр)        | 5. <b>Desktop</b> (Рабочий стол) |
| 3. <b>Debug</b> (Отладка)        | 6. <b>Internet</b> (Интернет)    |

Группа Standard	
	Открывает доступ к Репозиторию Объектов. Эквивалент опции File   New   Other (этой кнопки нет в версиях 1, 2 и 3)
	Открывает существующий файл. Эквивалент опции File   Open File
	Сохраняет файл на диске. Эквивалент опции File   Save File (клавиши быстрого доступа Ctrl-S)
	Сохраняя все файлы проекта. Эквивалент опции File   Save All
	Открывает созданный ранее проект программы. Эквивалент опции File   Open Project (клавиши быстрого доступа Ctrl-F11)
	Добавляет новый файл к проекту. Эквивалент опции Project   Add to project (клавиши быстрого доступа Shift-F11)
	Удаляет файл из проекта. Эквивалент опции Project   Remove from Project

Группа View	
	Выбирает модуль из списка модулей, связанных с текущим проектом. Эквивалент опции View   units (клавиши быстрого доступа Shift-F12)
	Выбирает форму из списка форм, связанных с текущим проектом. Эквивалент опции View   Forms (клавиши быстрого доступа Ctrl-F12)
	Переключает активность между окном формы и окном кода программы. Эквивалент опции View   Toggle Form/Unit (клавиша быстрого доступа F12)
	Создает новую форму и добавляет ее к проекту. Эквивалент опции File   New   Form (этой кнопки нет в версиях 1 и 2)

Группа Debug	
	Компилирует и выполняет программу. Эквивалент опции Run   Run (клавиша быстрого доступа F9)
	Реализует паузу в работе отлаживаемой программы. Эквивалент опции Run   Program Pause
	Осуществляет пошаговую трассировку программы с прослеживанием работы вызываемых подпрограмм. Эквивалент опции Run   Trace into (клавиша быстрого доступа F7)
	Осуществляет пошаговую трассировку программы, но не прослеживает работу вызываемых подпрограмм. Эквивалент опции Run   Step Over (клавиша быстрого доступа F8)

Группа View	
	Выбирает модуль из списка модулей, связанных с текущим проектом. Эквивалент опции View   units (клавиши быстрого доступа Shift-F12)
	Выбирает форму из списка форм, связанных с текущим проектом. Эквивалент опции View   Forms (клавиши быстрого доступа Ctrl-F12)
	Переключает активность между окном формы и окном кода программы. Эквивалент опции View   Toggle Form/Unit (клавиша быстрого доступа F12)
	Создает новую форму и добавляет ее к проекту. Эквивалент опции File   New   Form (этой кнопки нет в версиях 1 и 2)

Группа Debug	
	Компилирует и выполняет программу. Эквивалент опции Run   Run (клавиша быстрого доступа F9)
	Реализует паузу в работе отлаживаемой программы. Эквивалент опции Run   Program Pause
	Осуществляет пошаговую трассировку программы с прослеживанием работы вызываемых подпрограмм. Эквивалент опции Run   Trace into (клавиша быстрого доступа F7)
	Осуществляет пошаговую трассировку программы, но не прослеживает работу вызываемых подпрограмм. Эквивалент опции Run   Step Over (клавиша быстрого доступа F8)

Главное меню (Помимо главного меню в Delphi широко используется система вспомогательных меню, доступ к которым осуществляется правой кнопкой мыши.) содержит все необходимые средства для управления проектом. Все опции главного меню представляют собой опции-заголовки, открывающие доступ к выпадающим меню второго уровня.

Окно формы представляет собой проект Windows-окна будущей программы. Вся рабочая область окна обычно заполнена точками координатной сетки, служащей для упорядочения размещаемых на форме компонентов.

Окно дерева объектов появилось в версии 6 и предназначено для наглядного отображения связей между отдельными компонентами, размещенными на активной форме или в активном модуле данных.

Любой размещаемый на форме компонент характеризуется некоторым набором параметров: положением, размером, цветом и т. д. Часть этих параметров, например, положение и размеры компонента, программист может изменять, манипулируя с компонентом в окне формы. Для изменения других параметров предназначено окно Инспектора объектов. Это окно содержит две страницы - Properties (Свойства) и Events (События). Страница properties служит для установки нужных свойств компонента, страница Events позволяет определить реакцию компонента на то или иное событие. Совокупность свойств отображает видимую сторону компонента: положение относительно левого верхнего угла рабочей области формы, его размеры и цвет, шрифт и текст надписи на нем и т. п.; совокупность событий - его поведенческую сторону: будет ли компонент реагировать на щелчок мыши или на нажатие клавиш, как он будет вести себя в момент появления на экране или в момент изменения размеров окна и т. п.

Окно кода предназначено для создания и редактирования текста программы. Этот текст составляется по специальным правилам и описывает алгоритм работы программы. Совокупность правил записи текста называется языком программирования. В системе Delphi используется язык программирования Object Pascal, который представляет собой расширенную и усовершенствованную версию широко распространенного языка Паскаль, впервые предложенного швейцарским ученым Н. Виртом еще в 1970 г. и усовершенствованного сотрудниками корпорации Borland (созданные ими языки назывались Turbo Pascal, Borland Pascal и Object Pascal).

Первоначально окно кода содержит минимальный исходный текст, обеспечивающий нормальное функционирование пустой формы в качестве

полноценного Windows-окна. В ходе работы над проектом программист вносит в него необходимые дополнения, чтобы придать программе нужную функциональность.

### **Практическая часть.**

#### **Организация проекта в Delphi, основные файлы проектов**

Проект Delphi состоит из форм, модулей, установок параметров проекта, ресурсов и т.д. Вся эта информация размещается в файлах. Многие из этих файлов автоматически создаются Delphi, когда вы строите ваше приложение. Ресурсы, такие как битовые матрицы, пиктограммы и т.д., находятся в файлах, которые вы получаете из других источников или создаете при помощи многочисленных инструментов и редакторов ресурсов, имеющихся в вашем распоряжении. Кроме того, компилятор также создает файлы.

Когда вы проектируете приложение, Delphi создает следующие файлы:

Файл проекта (.dpr) - этот текстовый файл используется для хранения информации о формах и модулях. В нем содержатся операторы инициализации и запуска программ на выполнение

Файл модуля (.pas) - каждой создаваемой вами форме соответствует текстовый файл модуля, используемый для хранения кода. Можно создавать модули, не связанные с формами. Многие из функций и процедур Delphi хранятся в модулях.

Файл формы (.dfm) - это двоичный или текстовый файл, который создается Delphi для хранения информации о ваших формах. Каждому файлу формы соответствует файл модуля (.pas).

Файл параметров проекта(.dfo) - в этом файле хранятся установки параметров проекта.

Файл ресурсов(.res) - этот бинарный файл содержит используемую проектом пиктограмму и прочие ресурсы.

Файлы резервных копий (.~dp, .~df, .~pa) - это соответственно файлы резервных копий для файлов проекта, формы и модуля. Если что-то безнадежно испорчено в проекте, можно соответственно изменить расширения этих файлов и таким образом вернуться к предыдущему не испорченному варианту.

Файл конфигурации окон (.dsk) - файл хранит конфигурацию всех окон среды разработки.

Исполняемый файл (.exe) - это исполняемый файл вашего приложения. Он является автономным исполняемым файлом, для которого больше ничего не требуется, если только вы не используете библиотеки, содержащиеся в DLL, OCX и т.д., а также если вы не используете поддержку пакетов времени выполнения.

Объектный файл модуля (.dcu) - Это откомпилированный файл модуля (.pas), который компонуется в окончательный исполняемый файл.

И, наконец, другие файлы Windows, которые могут использоваться Delphi:

Файлы справки (.hlp) - это стандартные файлы справки Windows, которые могут быть использованы вашим приложением Delphi.

Файлы изображений или графические файлы (.wmf, .bmp, .ico).

Эти файлы обычно используются в приложениях Windows для создания привлекательного и дружелюбного пользовательского интерфейса.

Главной частью вашего приложения является файл проекта (.dpr), содержащий код на языке Object Pascal, с которого начинается выполнение вашей программы и который обеспечивает инициализацию других модулей. Он создается и модифицируется Delphi автоматически в процессе разработки приложения. Имя, которое вы даете файлу проекта, когда сохраняете его, становится именем исполняемого файла.

### **Контрольные вопросы**

1. Каково назначение и возможности среды визуального программирования Borland Delphi?
2. Назначение окон Borland Delphi.
3. Какого назначения палитры компонент?
4. Назначение пиктографических кнопок среды Borland Delphi.
5. Описать окно кода программы и инспектора объектов.

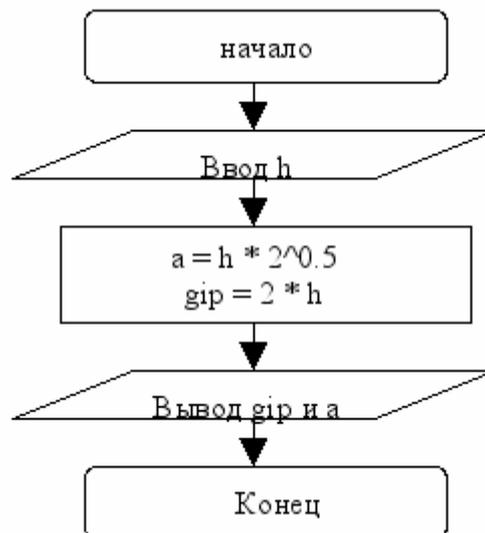
**Задание:** Решить следующие задачи по образцу:

#### Теоретический материал

При линейном программировании одни операторы идут строго за другими. На практике редко встречаются задачи, алгоритм решения которых является линейным. Часто оказывается, что алгоритм решения даже элементарной задачи не является линейным. Например, пусть надо вычислить по формуле ток в электрической цепи. Если предположить, что пользователь всегда будет вводить верные данные, то алгоритм решения этой задачи действительно является линейным. Однако полагаться на то, что пользователь будет вести себя так, как надо программе, не следует. Формула расчета предполагает, что величина сопротивления не равна нулю. А что будет, если пользователь введет 0? Ответ простой: возникнет ошибка "Деление на ноль", и программа аварийно завершит работу. Можно, конечно, возложить ответственность за это на пользователя, но лучше внести изменения в алгоритм решения, чтобы расчет выполнялся только в том случае, если введены верные данные.

#### Например:

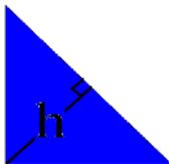
Задача № 1. В равнобедренном прямоугольном треугольнике известна высота  $h$ , опущенная на гипотенузу. Найти стороны треугольника.



Описание интерфейса:

The screenshot shows a window titled "Form1" with a light gray background. At the top, there is a text label: "В равнобедренном прямоугольном треугольнике известна высота h, опущенная на гипотенузу. Найти стороны треугольника." Below this text is a blue right-angled isosceles triangle with a dashed line representing the height 'h' from the right angle to the hypotenuse. To the right of the triangle is a text label "Введите длину высоты:" followed by an empty text input field. Below the input field is a button labeled "Вычислить длины сторон треугольника". At the bottom left, there are two labels: "Длина катетов:" and "Длина гипотенуза:". At the bottom right, there is a button labeled "Close" with a small icon.

В таблице показаны свойства объектов

Объект	Свойство	Устанавливаемое значение
Label1	Caption	В равнобедренном прямоугольном треугольнике известна высота $h$ , опущенная на гипотенузу. Найти стороны треугольника.
Image1	Picture	
Label2	Caption	Введите длину высоты:
Edit1	Text	“ ”
Button1	Caption	Вычислить длины сторон треугольника
	WordWrap	True
Label3	Caption	Длина катетов:
Label4	Caption	Длина гипотенуза:
BitBtn1	Kind	bkClose

### Текст программы:

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, Buttons;
type
  TForm1 = class(TForm)
    Label1: TLabel;
    Image1: TImage;
    BitBtn1: TBitBtn;
    Label2: TLabel;
    Edit1: TEdit;
    Button1: TButton;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
  procedure Button1Click(Sender: TObject);
  end;

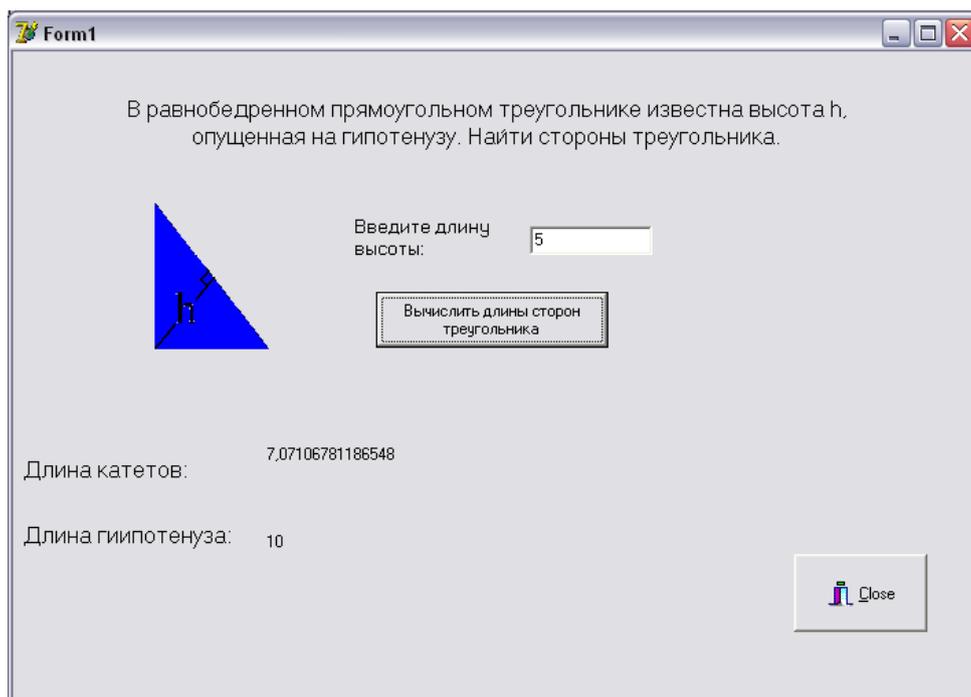
```

```

private
{ Private declarations }
public
{ Public declarations }
end;
var
  Form1: TForm1;
implementation
{$R *.dfm}
var h, a, gip : real;
var a1,gip1: string;
procedure TForm1.Button1Click(Sender: TObject); // Щелчек по командной
кнопке
begin //Начать работу
  h := StrToFloat(Form1.Edit1.Text); //Высоту h берем из текстового поля и
переводим в вещественный тип
  a := h*sqrt(2); //Вычисляем длину катета
  gip:= 2*h; // ычисляем длину гипотенузы
  a1:=FloatToStr(a); // Переводим вещественные величины в строковые
  gip1:=FloatToStr(gip); // Переводим вещественные величины в
строковые
  Form1.Label5.Caption := a1; //Выводи результат на форму
  Form1.Label6.Caption := gip1 //Выводи результат на форму
end;
end.

```

### Тестирование:



### Задания для самостоятельного выполнения:

Создать в соответствии с вариантом задания программу линейной структуры. В программе должны быть использованы 2-3 визуальных компонента (выбираются самостоятельно студентом). Поработать со свойствами компонент и формы.

**Задание 1.** Вычислить значения переменных, указанных в таблице ниже (вариант задается преподавателем), по заданным расчетным формулам и наборам исходных данных. Вывести на форму значения вводимых исходных данных и результаты вычислений, сопровождая вывод наименованиями выводимых переменных.

Вариант задания	Расчетные формулы	Значения исходных данных
1	$c = \left  x^{y/x} - \sqrt[3]{y/x} \right ; \quad b = \sqrt[3]{-x} \frac{y-z/\sqrt[3]{-x}}{1+\sqrt[3]{-x}}$	$x=1,825$ $y=18,225$ $z=-3,298$
2	$s = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}; \quad t = x \left( \sin x^3 + \cos^2 y \right)$	$x=0,335$ $y=0,025$
3	$y = e^{bt} \sin \left( \sqrt{bt+a} \right); \quad s = b \sin \left( t^2 \cos 2t \right) - 1$	$a=-0,5$ $b=1,7$ $t=0,44$
4	$g = \sqrt{x^2+b} - b^2 \sin^3 \left( \sqrt{a} \right) x; \quad y = \cos^2 x^3 - x/\sqrt{a^2+b^2}$	$a=1,5$ $b=15,5$ $x=-2,9$
5	$s = x^3 t g^2 \left( \sqrt{b} \right) + a/\sqrt{x+b}; \quad q = \frac{bx^2 - a}{e^{ax} - 1}$	$a=16,5$ $b=3,4$ $x=0,61$
6	$R = x^2 \left( \sqrt{b} - \sin^2 \left( \sqrt{a} \right) \right); \quad y = \frac{2 \cos \left( -\pi/6 \right)}{1/2 + \sin^2 a}$	$a=0,7$ $b=0,05$ $x=0,5$
7	$y = \sin^3 \left( \sqrt{a} \right) - \sqrt{x/b}; \quad z = \frac{x^2}{a} + \cos \left( \sqrt{b} \right)$	$a=1,1$ $b=0,004$ $x=0,2$
8	$f = \sqrt[3]{mgt +  c \sin t }; \quad z = m \cos \left( t \sin t \right) + c$	$m=2$ $c=-1$ $t=1,2$ $b=0,7$
9	$y = bt g^2 x - \frac{a}{\sin^2 \left( \sqrt{a} \right)}; \quad d = a e^{\sqrt{a}} \cos \left( x/a \right)$	$a=3,2$ $b=17,5$ $x=-4,8$
10	$f = \ln \left( \sqrt{1+x^2} \right) + \sin^2 \left( \sqrt{b} \right); \quad z = e^{cx} \frac{x + \sqrt{x+a}}{x - \sqrt{ x-b }}$	$a=10,2$ $b=9,2$ $x=2,2$ $c=0,5$

**Задание 2.** Найти значение выражения для вводимых с клавиатуры  $a$ ,  $b$ ,  $c$ .

Вариант задания	Расчетные формулы	Вариант задания	Расчетные формулы
1	$\operatorname{tg} \left( \frac{\operatorname{tg} \left( \frac{\operatorname{ctg}(a+b)}{\ln(\sqrt{c}+2)} \right)}{\operatorname{tg} \left( \frac{\sin a \cdot \sin b}{\sqrt{\cos c} \cdot a^2} \right)} \right) + \ln \left( \frac{\ln \left( \frac{\sqrt{a} + \sqrt{c}}{\sqrt{c} + \sqrt{b}} \right)}{\operatorname{ctg} \left( \frac{\sin \sqrt{c}}{\sin \sqrt{b}} \right)} \right)$ <p>при <math>a=4, b=2, c=5</math> ответ -0.86739287; при <math>a=4.1, b=3.2, c=1</math> ответ -2.4316158</p>	6	$\frac{a \cdot c - \sqrt{a^2 + \sqrt{b} - \cos(a^3)}}{\frac{\ln(5a)}{\sin(6b)} + \frac{\operatorname{ctg}(a-b)}{e^{ca}}} + \sqrt{\frac{a + \sqrt{\ln(5/b) + 6}}{a^2 + \frac{a+b^4-c}{5 + \frac{a}{c-b}}}}$ <p>при <math>a=5, b=2, c=4</math> ответ -1.9610729; при <math>a=2.3, b=1, c=4</math> ответ 0.20184899</p>
2	$e^{\frac{\operatorname{tga}+b}{\sin a - \ln c}} \cdot \operatorname{tg} \frac{\sqrt{a+\sqrt{c}}}{\cos \sqrt{a}} \cdot \ln \frac{\sqrt{a+b} - c}{\sqrt{a+\sqrt{b}}}$ $c + \frac{c + \sqrt{5a}}{c + \frac{\sqrt{a+b}}{b\sqrt{c}}}$ <p>при <math>a=3, b=2, c=1</math> ответ -83595.544; при <math>a=0.1, b=0.2, c=0.3</math> ответ -0.67568268</p>	7	$\cos \left( e^{\frac{\operatorname{tg} a}{b+c}} \right) + \sin \left( e^{\frac{\operatorname{ctg} b}{a} + 2} \right)$ $\sqrt{\frac{\operatorname{tg} \frac{a-3}{2b} \cdot \operatorname{ctg} \frac{b+4}{2c}}{2+4c^2 - \sqrt{b} + \cos c}} + \frac{\ln c}{\operatorname{tg}(a-b)} + 5$ $\operatorname{tg} \frac{a}{\sqrt{c}} \cdot \operatorname{ctg} \frac{a+b}{c}$ <p>при <math>a=3, b=1, c=4</math> ответ 1.0275070; при <math>a=-1, b=2.1, c=5</math> ответ -0.00192481</p>
3	$\frac{a \cdot b}{c + \sin c} + \ln \frac{\ln a}{\sin b} - \sqrt{a^2 - b^2} \cos c$ $\frac{\sqrt{\frac{a-5}{c+a}} - \sqrt{\sqrt{a+\sqrt{c}} + \sqrt{2 \sin \sqrt{a}}}}{\sqrt{c+a}}$ <p>при <math>a=7, b=2, c=1</math> ответ -1.2678855; при <math>a=7.1, b=2, c=3</math> ответ 1.61379036</p>	8	$\operatorname{tg} \frac{\sin a}{\sin b} + e^{\operatorname{ctg}(\cos b)} - \frac{a}{b \cdot c} + c^3$ $\cos \left( a + \frac{\cos \left( b + \frac{\cos(c + \ln a)}{\sin(a + \ln b)} \right)}{\sqrt{c^3 + \operatorname{ctg}(a-b) \cdot \sin \frac{a}{b}}} \right)$ <p>при <math>a=1, b=2, c=3</math> ответ 62.828741; при <math>a=1.2, b=2.3, c=4</math> ответ 176.09174</p>
4	$\frac{\sqrt{1 + \operatorname{tg}^2 a + b \operatorname{ctg}^2 c} - \sqrt{3 + \ln a + \sin b}}{2 \cdot \operatorname{tg} \frac{a}{b} + 3}$ $\frac{\operatorname{tga} + \operatorname{ctg} \frac{a}{c}}{\sqrt{\ln \left( \frac{a+b}{c} \right)}} + \sqrt{\frac{\cos a \cdot \cos b}{\ln \left( \frac{a+b}{c} \right)}} - \sin \frac{a+b}{c}$ <p>при <math>a=3, b=2, c=1</math> ответ 0.30388583; при <math>a=1, b=2.3, c=4</math> ответ 0.24574518</p>	9	$\operatorname{tg} \frac{\sqrt{a \cdot b}}{\sin(c+b)} - 8b + (a^2 + b^2)^3 + \frac{8}{9} ca$ $\frac{c}{a} \sqrt{\frac{a^2}{4} - b \cdot c + \sqrt{c} - \ln(b+c)} + \frac{8}{9} ca (8+a)^{3/2}$ <p>при <math>a=3, b=2, c=1</math> ответ 16718.692; при <math>a=0.1, b=0.2, c=0.3</math> ответ -0.3959739</p>

Вариант задания	Расчетные формулы	Вариант задания	Расчетные формулы
5	$\sqrt{3 + \sqrt{a - \frac{c + \sqrt{a} \cdot \sin \sqrt{c}}{\sin(3b) \cdot \cos(3b)}}}$ $\sqrt{8 - \sqrt{b + \frac{\ln(1 + \sin a)}{\sqrt{\frac{b+c}{a^2-1} + \frac{a}{c \cdot b}}}}}$ <p>при <math>a=3, b=2, c=1</math> ответ 0.9941356; при <math>a=3, b=4.1, c=5</math> ответ 1.1810022.</p>	10	$\frac{\sqrt{\sin^2(a^3) + \cos^2(b^3)}}{\sqrt{abc}}$ $\ln\left(\left(\frac{a}{b} + \frac{b}{c} + \frac{c}{a}\right)^{\frac{3}{2}} - \frac{(ac)^3}{b^2}\right)$ <p>при <math>a=3, b=7, c=2</math> ответ 0.11616104; при <math>a=0.1, b=0.2, c=0.3</math> ответ 6.0308170</p>

### Задание 3

1. Дано четырехзначное число. а) Получить число, образуемое при перестановке первой и второй, третьей и четвертой цифр этого числа. б) Получить число, образуемое при перестановке второй и третьей цифр этого числа.
2. Дано трехзначное число. Найти число, образованное при перестановке цифр единиц и сотен этого числа.
3. Дано трехзначное число. Найти: число единиц этого числа (младший десятичный разряд); число десятков; число сотен; сумму и произведение цифр этого числа.
4. Дано трехзначное число. Найти число, образованное при прочтении его цифр справа налево.
5. Дано трехзначное число. В нем зачеркнули первую цифру и приписали ее в конце. Найти полученное число.
6. Дано четырехзначное число. а) Найти число, полученное при прочтении его цифр справа налево. б) Получить число, образуемое при перестановке двух первых и двух последних цифр этого числа.
7. Дано трехзначное число. Найти число, полученное при перестановке первой и второй цифр этого числа.
8. Дано трехзначное число. Найти число, полученное при перестановке второй и третьей цифр заданного числа.
9. Дано трехзначное число. Найти 6 чисел, образованных всевозможными перестановками цифр этого числа.
10. Дано четырехзначное число. Найти: сумму и произведение его цифр.

### Задание 4

1. Вычислите площадь поверхности и объем куба.
2. Найти площадь равнобедренной трапеции с основаниями  $a$  и  $b$  и углом при большем основании  $\alpha$ .
3. Найти среднее геометрическое и среднее арифметическое трех чисел.
4. Найти гипотенузу и площадь прямоугольного треугольника по двум данным катетам.

5. Вычислить площадь параллелограмма по двум сторонам и углу между ними, заданному в градусах.
6. Вычислить длины медиан треугольника, заданного длинами сторон.
7. Треугольник задан величинами своих углов и радиусом описанной окружности. Вычислить длины сторон треугольника.
8. Дана длина ребра куба. Найти объем куба и площадь его боковой поверхности.
9. Определить периметр правильного  $n$ -угольника, описанного около окружности радиусом  $r$ .
10. Три сопротивления  $R_1$ ,  $R_2$ ,  $R_3$  соединены параллельно. Найти сопротивление соединения.

## **Лабораторная работа № 2**

### *Основы визуального программирования*

**Цель:** изучить основы визуального программирования в среде Borland Delphi. Научиться пользоваться основными визуальными компонентами и их свойствами.

#### **Теоретическая часть**

Программирование в Delphi строится на тесном взаимодействии двух процессов: процесса конструирования визуального проявления программы (т. е. ее Windows-окна) и процесса написания кода, придающего элементам этого окна и программе в целом необходимую функциональность. Для написания кода используется окно кода, для конструирования программы - остальные окна Delphi и, прежде всего, окно формы.

Между содержимым окон формы и кода существует неразрывная связь, которая строго отслеживается Delphi. Это означает, что размещение на форме компонента приводит к автоматическому изменению кода программы и наоборот - удаление тех или иных автоматически вставленных фрагментов кода может привести к удалению соответствующих компонентов. Помня об этом, программисты вначале конструируют форму, размещая на ней очередной компонент, а уже только после этого переходят, если это необходимо, к писанию фрагмента кода, обеспечивающего требуемое поведение компонента в работающей программе.

Как уже говорилось, окно формы содержит проект Windows-окна программы. Важно помнить, что с самого начала работы над новой программой Delphi создает минимально необходимый код, обеспечивающий ее нормальное функционирование в Windows. Таким образом, простейшая программа готова сразу после выбора опции File | New | Application, и нам остается просто запустить программу.

#### **Практическая часть**

Попробуем модифицировать программу, например, изменим заголовок ее окна. По умолчанию заголовок окна совпадает с заголовком формы: Form1. Чтобы изменить заголовок, нужно обратиться к окну Инспектора

объектов. Закройте окно работающей программы Form1, щелкните мышью по строке Caption и введите название формы.

Итак, простым изменением содержимого строки в окне Инспектора объектов мы добились важной перемены: изменили одно из свойств окна программы - его заголовок. Таким же образом можно изменять любое другое свойство формы.

Разместить на форме новый компонент не представляет труда. Для этого сначала нужно его выбрать (щелкнуть по нему мышью) в палитре компонентов, а затем щелкнуть мышью по точке рабочего пространства формы, где должен располагаться левый верхний угол компонента.

Поработайте со свойствами компонент, вынесенных Вами на форму. Попробуйте реализовать реакцию на события данных компонент.

Таким образом:

- процесс создания Delphi-программы разбивается на две фазы: фазу конструирования формы и фазу кодирования;
- конструирование формы осуществляется с помощью выбора компонентов из палитры и размещения их на форме;
- программист может перемещать любой размещенный на форме компонент и изменять его размеры с помощью мыши;
- чтобы придать компоненту нужные свойства, используется страница Properties Инспектора объектов;
- чтобы компонент мог откликаться на то или иное событие, программист должен создать обработчик события и указать его имя на странице Events Инспектора объектов;
- обработчик события оформляется в виде процедуры, имеющей составное имя. Первая часть имени представляет собой имя класса для формы, вторая часть отделяется от первой точкой и может быть произвольной. Если Delphi автоматически формирует заготовку для обработчика, то вторая часть имени представляет собой объединение имени компонента и имени события без предлога On;
- тело процедуры ограничено словами begin... end и состоит из отдельных предложений (операторов) языка Object Pascal. В конце каждого предложения ставится точка с запятой;
- свойства компонента могут изменяться на этапе прогона программы.

### **Контрольные вопросы**

1. Поясните свойства и их назначение компонента Form.
2. Поясните свойства и их назначение компонентов Label, Edit, Memo, Button, CheckBox, RadioButton, GroupBox, RadioGroup, Panel.

### **Задание**

Создать в соответствии с вариантом задания программу. В программе должны быть использованы 4-5 визуальных компонентов (выбираются самостоятельно студентом). Поработать со свойствами компонент и формы.

Варианты заданий на структуру ветвление

Вариант задания	Функция	Условие	Исходные данные
1	$y = \begin{cases} at^2 \ln t \\ 1 \\ e^{at} \cos bt \end{cases}$	$\begin{cases} 1 \leq t \leq 2 \\ t < 1 \\ t > 2 \end{cases}$	$\begin{cases} a = -0.5 \\ b = 2 \end{cases}$
2	$y = \begin{cases} \pi x^2 - \frac{7}{x^2} \\ ax^3 + 7\sqrt{x} \\ \lg \left( \frac{1}{x} + 7\sqrt{x} \right) \end{cases}$	$\begin{cases} x < 1.3 \\ x = 3 \\ x > 1.3 \end{cases}$	$a = 1.5$
3	$w = \begin{cases} ax^2 + bx + c \\ \frac{a}{x} + \sqrt{x^2 + 1} \\ \frac{a + bx}{\sqrt{x^2 + 1}} \end{cases}$	$\begin{cases} x < 1.2 \\ x = 2 \\ x > 1.2 \end{cases}$	$\begin{cases} a = 2.8 \\ b = -0.3 \\ c = 4 \end{cases}$
4	$Q = \begin{cases} \pi x^2 - \frac{7}{x^2} \\ ax^3 + 7\sqrt{x} \\ \ln \left( \frac{1}{x} + 7\sqrt{ x+a } \right) \end{cases}$	$\begin{cases} x < 1.4 \\ x = 1.4 \\ x > 1.4 \end{cases}$	$a = 1.65$
5	$y = \begin{cases} 1.5 \cos^2 x \\ 1.8ax \\ \left( \frac{1}{x} - 2 \right)^2 + 6 \\ 3 \operatorname{tg} x \end{cases}$	$\begin{cases} x < 1 \\ x = 1 \\ 1 < x < 2 \\ x > 2 \end{cases}$	$a = 2.3$
6	$y = \begin{cases} x\sqrt[3]{x-a} \\ x \sin ax \\ e^{-ax} \cos ax \end{cases}$	$\begin{cases} x > a \\ x = a \\ x < a \end{cases}$	$a = 2,5$
7	$q = \begin{cases} bx - \lg bx \\ 1 \\ bx + \lg bx \end{cases}$	$\begin{cases} bx < 1 \\ bx = 1 \\ bx > 1 \end{cases}$	$b = 1,5$
8	$t = \begin{cases} \sin y \lg x \\ \cos^2 x \end{cases}$	$\begin{cases} x > 3,5 \\ x \leq 3,5 \end{cases}$	$y = 0,5$
9	$f = \begin{cases} \lg \left( \frac{1}{x} + 1 \right) \\ \sin^2 \sqrt{ ax } \end{cases}$	$\begin{cases} x > 1 \\ x \leq 1 \end{cases}$	$a = 20,3$

Вариант задания	Функция	Условие	Исходные данные
10	$z = \begin{cases} \sqrt[3]{x+x^2} \sqrt{x+t} \\ \sqrt{x+t} + 1/x \\ \cos x + t \sin^2 x \end{cases}$	$x < 0,5$ $x = 0,5$ $x > 0,5$	$t = 2,2$

### Задание 2

1. Если треугольник со сторонами  $a, b, c$  является равнобедренным, найти его площадь, в противном случае найти его периметр.
2. Даны действительные положительные числа  $a, b, c, x$  и  $y$ . Выясните, пройдет ли кирпич с ребрами  $a, b, c$  в прямоугольное отверстие со сторонами  $x$  и  $y$ . Просовывать кирпич в отверстие разрешается только так, чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия.
3. Написать программу, которая присваивает переменной  $k$  значение квадрата исходного данного, если  $a$  – нечетное число, и значение куба, если  $a$  – четное.
4. Если треугольник со сторонами  $a, b, c$  ( $a \leq b \leq c$ ) является прямоугольным, то вычислить длину медианы, выходящей из прямого угла, в противном случае вычислить его периметр.
5. Числа  $a$  и  $b$  выражают длины катетов одного прямоугольного треугольника, числа  $c$  и  $d$  – другого. Определить, являются ли треугольники подобными.
6. Длины сторон выпуклого четырехугольника выражаются числами  $a, b, c, d$ . Определить, можно ли в него вписать окружность.
7. Задано круглое отверстие радиусом  $r$ . Пройдет ли через него кирпич с длинами сторон  $a, b, c$ ?
8. Для трех чисел  $A, B, C$  найти произведение двух наибольших, пользуясь формулой  $P = \frac{ABC}{\min(A, B, C)}$ .
9. Составьте программу, определяющую, лежит ли точка с указанными координатами  $X, Y$  на окружности радиуса  $R$  с центром в начале координат.
10. Составьте программу, которая по трем введенным вами числам определит, могут ли эти числа быть длинами сторон треугольника, и если да, то какой получится треугольник с данными длинами сторон (прямоугольный, остроугольный, тупоугольный).

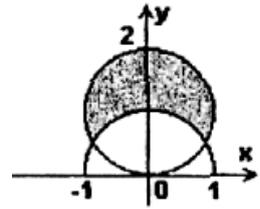
**Задание 3.** Вывести на экран значения исходных данных и результатов промежуточных и окончательных вычислений, сопровождая вывод именами переменных (таблица).

Вариант задания	Расчетные формулы	Вариант задания	Расчетные формулы
1	$y = \operatorname{ctgx} - \sqrt{1+x^2}, \text{ где}$ $x = \frac{z_1^3 - 1}{z_2^2 + 1}; z_1 = \frac{90 - b^2}{\sqrt{3b^2 + 1}};$ $z_2 = \begin{cases} \operatorname{tgb}, & \text{если } b = 10 \\ 1, & \text{если } b < 10 \end{cases}$ <p>значение <math>b</math> задается</p>	6	$y = \frac{a}{2}(e^{\frac{x}{2}} + e^{\frac{x}{2}}), z = \frac{f_1^3 + f_2}{\lg 2f_2 },$ $f_1 = 2^{\frac{1}{k-2}}, f_2 = \sqrt{ k } \sin x,$ $x = \begin{cases} k+1, & \text{если } k > 0 \\ k-1, & \text{если } k \leq 0 \end{cases}$ <p>значения <math>k</math> и <math>a</math> задаются</p>
2	$y = e^x \sqrt{1+e^{2x}} + \operatorname{arctge}^2, \text{ где}$ $x = z_2 \cos^2 z_1 + \sin a^2; z_1 = \frac{3a}{1-2a};$ $z_2 = \begin{cases} a^2, & \text{если } a > 0 \\ 0, & \text{если } a \leq 0 \end{cases}$ <p>значение <math>a</math> задается</p>	7	$y = \sqrt{z_1} \frac{\cos x}{\sin^2 x} + \ln \left  \operatorname{tg} \frac{x}{2} \right ;$ $x = \frac{1,5z - 5}{3z + 5}; z = \frac{e^{-3z_1} - 1}{z_2}$ $z_1 = \begin{cases}  z_2 , & \text{если } z_2 < 0 \\ \frac{z_2}{8}, & \text{если } z_2 > 0 \end{cases}$ <p>значение <math>z_2</math> задается</p>
3	$y = x^2 + b - \sqrt{b^2} \sin b, \text{ где}$ $x = \frac{\operatorname{tg} z_1}{z_2}; z_1 = \frac{\sin^2 \frac{a}{b}}{a^2};$ $z^2 = \begin{cases} ab, & \text{если } a < 0, b > 0 \\ 1, & \text{если } a < 0, b < 0 \end{cases}$ <p>значения <math>a</math> и <math>b</math> задаются</p>	8	$f = (z+1) \operatorname{arctg} 2z,$ $z = \frac{x_1^2 - x_1 - 2}{x_2^3 + 1}, x_1 = \frac{e^{2x_2}}{\sin(2y)},$ $x_2 = \begin{cases} \sin y, & \text{если } 100 \leq y \leq 200 \\ \cos y, & \text{если } y < 100 \end{cases}$ <p>значение <math>y</math> задается</p>
4	$y = \operatorname{arctgx} + \ln \sqrt{\frac{1+x}{1-x}} + e^x, \text{ где}$ $x = \frac{1 - \sin m}{a_1^2}; a_1 = \frac{\cos \frac{b}{a_2}}{b}$ $a_2 = \begin{cases}  b , & \text{если } b < 0 \\ 2, & \text{если } b > 0 \end{cases}$ <p>значения <math>b</math> и <math>m</math> задаются</p>	9	$y = \operatorname{ctgx} - \sqrt{1+x^2}, \text{ где}$ $x = \frac{z_1^3 - 1}{z_2^2 + 1}; z_1 = \frac{90 - b^2}{\sqrt{3b^2 + 1}};$ $z_2 = \begin{cases} \operatorname{tgb}, & \text{если } b = 10 \\ 1, & \text{если } b < 10 \end{cases}$ <p>значение <math>b</math> задается</p>
5	$y = x \operatorname{arctg} \frac{x}{a} - \frac{a}{2} \ln(x^2 + a^2), \text{ где}$ $a = \frac{0,01 \cdot (z_1^3 + 1)}{\operatorname{tg}(z_2^2 + 1)}; z_2 = \frac{\sqrt{1 - \cos^2 2c}}{(\cos 2c)};$ $z_1 = \begin{cases} \sqrt{c}, & \text{если } c > 0 \\ -c, & \text{если } c \leq 0 \end{cases}$ <p>значения <math>c</math> и <math>x</math> задаются</p>	10	$y = e^x \sqrt{1+e^{2x}} + \operatorname{arctge}^2, \text{ где}$ $x = z_2 \cos^2 z_1 + \sin a^2; z_1 = \frac{3a}{1-2a};$ $z_2 = \begin{cases} a^2, & \text{если } a > 0 \\ 0, & \text{если } a \leq 0 \end{cases}$ <p>значение <math>a</math> задается</p>

### Задание 4

1.  $D$  - заштрихованная часть плоскости. Пусть  $F$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

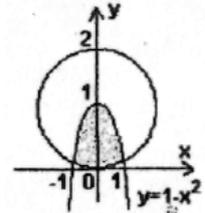
$$F = \begin{cases} (x-y)^3, & \text{если } (x, y) \in D, \\ 0 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $F$ .

2.  $D$  - заштрихованная часть плоскости. Пусть  $M$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

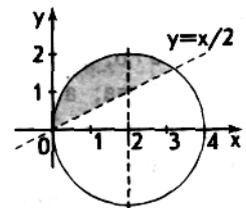
$$M = \begin{cases} 2x + y^2, & \text{если } (x, y) \in D, \\ y^2 - 2x & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $M$ .

3.  $D$  - заштрихованная часть плоскости. Пусть  $U$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

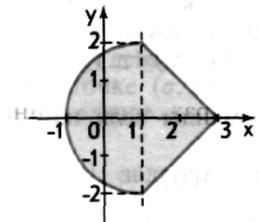
$$U = \begin{cases} x^2(x+y), & \text{если } (x, y) \in D, \\ 5.4 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $U$ .

4.  $D$  - заштрихованная часть плоскости. Пусть  $L$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

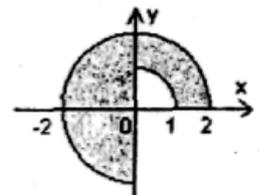
$$F = \begin{cases} \frac{xy}{4}, & \text{если } (x, y) \in D, \\ -3 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $L$ .

5.  $D$  - заштрихованная часть плоскости. Пусть  $N$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

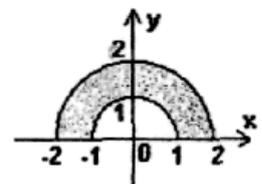
$$N = \begin{cases} x^3(y+1), & \text{если } (x, y) \in D, \\ 2x & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $N$ .

6.  $D$  - заштрихованная часть плоскости. Пусть  $R$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

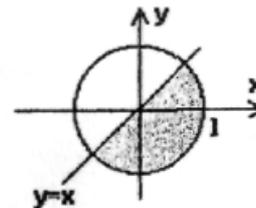
$$R = \begin{cases} \cos^2(x+y), & \text{если } (x, y) \in D, \\ xy & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $R$ .

7.  $D$  - заштрихованная часть плоскости. Пусть  $K$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

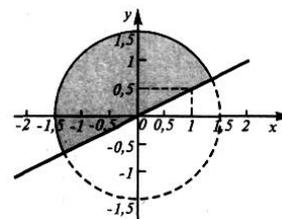
$$K = \begin{cases} \frac{x^2 + y^2}{x + y}, & \text{если } (x, y) \in D, \\ 0 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $K$ .

8.  $D$  - заштрихованная часть плоскости. Пусть  $Z$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

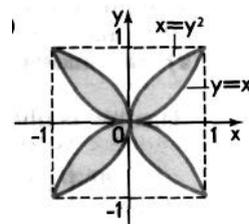
$$Z = \begin{cases} \frac{x^3 + y^2}{y^3 + x^2}, & \text{если } (x, y) \in D, \\ -1 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $Z$ .

9.  $D$  - заштрихованная часть плоскости. Пусть  $S$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

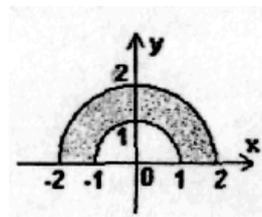
$$S = \begin{cases} \sin^2 x - y, & \text{если } (x, y) \in D, \\ 0 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $S$ .

10.  $D$  - заштрихованная часть плоскости. Пусть  $T$  определяется по  $x$  и  $y$  следующим образом (запись  $(x, y) \in D$  означает, что точка с координатами  $x, y$  принадлежит  $D$ ):

$$T = \begin{cases} x^2 - y, & \text{если } (x, y) \in D, \\ y^2 & \text{в противном случае.} \end{cases}$$



Даны действительные числа  $x, y$ . Определить  $T$ .

### Варианты заданий на структуру Цикл

#### **Задание 1.**

1. Найти сумму ряда:  $S = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots + \frac{1}{n^2}$  с заданной точностью  $\varepsilon = 10^{-3}$ .
2. Найти сумму ряда:  $S = \frac{1}{1+1^2} + \frac{2}{1+2^2} + \frac{3}{1+3^2} + \dots$  с заданной точностью  $\varepsilon = 10^{-3}$ .
3. Найти сумму 35 членов ряда:  $S = \frac{2}{3} + \frac{4}{9} + \frac{6}{27} + \frac{8}{81} + \dots$ .
4. Найти сумму 50 членов ряда:  $S = 2 + \frac{2}{2 \cdot 3} + \frac{3}{3^2 \cdot 5} + \frac{2}{2^3 \cdot 7} + \dots$ .
5. Найти сумму ряда:  $S = \frac{1}{2^2} + \frac{3}{4^2} + \frac{5}{6^2} + \frac{7}{8^2} + \frac{9}{10^2} + \dots$  с заданной точностью  $\varepsilon = 10^{-3}$ .
6. Найти сумму ряда:  $S = \frac{1}{1+1!} + \frac{2}{1+2!} + \frac{3}{1+3!} + \dots$  с заданной точностью  $\varepsilon = 10^{-4}$ .

7. Найти сумму ряда:  $\ln(2) \cong S = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots + (-1)^{n+1} \frac{1}{n}$  с заданной точностью  $\varepsilon = 10^{-4}$ .
8. Найти сумму ряда:  $S = -\frac{(2x)^2}{2} + \frac{(2x)^4}{24} + \dots + (-1)^n \frac{(2x)^{2n}}{(2n)!}$  при  $x=0,10$  с заданной точностью  $\varepsilon = 10^{-9}$ .
9. Найти сумму ряда:  $S = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots + (-1)^{n+1} \frac{x^{2n-1}}{2n-1}$  при  $x=0,20$  с заданной точностью  $\varepsilon = 0,5 \cdot 10^{-4}$ .
10. Найти сумму ряда, начиная отсчет с нулевого члена:  
 $S = 1 + \frac{x^2}{2!} - \frac{3x^4}{4!} + \dots + (-1)^n \frac{2n-1}{(2n)!} x^{2n}$  при  $x=0,75$  с заданной точностью  $\varepsilon = 10^{-3}$ .

### Задание 2.

1. Дано натуральное число  $N$ . Вычислить:  $\frac{\sin 1}{\cos 1} \cdot \frac{\sin 2}{\cos 2} \cdot K \frac{\sin N}{\cos N}$ . При  $N=52$  должно получиться  $-7.4803638$ .
2. Дано натуральное число  $N$ . Вычислить:  $\frac{\sin 1}{\cos 1} \cdot \frac{\sin 1 + \sin 2}{\cos 1 + \cos 2} \cdot K \frac{\sin 1 + K + \sin N}{\cos 1 + K + \cos N}$ . При  $N=50$  должно получиться  $-3.7894719$ .
3. Даны натуральные числа  $n$  и  $k$ . Вычислить:  $\frac{n!}{k!(n-k)!}$ . При  $n=15, k=7$  должно получиться  $6435$ , при  $n=615, k=613$  должно получиться  $188805$ .
4. Даны: действительное число  $a$  и натуральное  $N$ . Найти:  $\frac{1}{a^0} + \frac{1}{a^2} + \frac{1}{a^4} + \dots + \frac{1}{a^{2n}}$ . При  $a=-1.2, n=4$  должно получиться  $2.7441635$ .
5. Дано действительное число  $x$  и натуральное число  $n$ . Среди чисел  $x^1, x^2, \dots, x^n$  найти наибольшее и наименьшее.  
 При  $x=-2.3, n=5$  —  $\max=27.9841, \min=-64.36343$ .
6. Даны: натуральное число  $k$  и вещественное  $a$ . Найти  $\frac{(a-1)(a-3) \cdot K \cdot (a-(2k+1))}{(a+2)(a+4) \cdot K \cdot (a+(2k+2))}$ .  
 При  $k=2, a=3.1$  должно получиться  $-0.00121089$ , при  $k=7, a=-77$  должно получиться  $5.99983188$ .
7. Даны: натуральное число  $n$  и действительное  $x$ . Найти среднее арифметическое чисел:  $\frac{x+1}{x-1}, \frac{x+2}{x-2}, \frac{x+4}{x-4}, k, \frac{x+2^n}{x-2^n}$ .  
 При  $n=6, x=16.5$  должно получиться  $9.58621117$ , при  $n=15, x=0$  должно получиться  $-1$ .
8. Дано натуральное число  $n$ . Вычислить  $1 \cdot 2 + 2 \cdot 3 \cdot 4 + \dots + n(n+1) \cdot \dots \cdot 2n$ . При  $n=8$  должно получиться  $4276578146$ .

9. Дано вещественное число  $a$ . Найти минимальное натуральное  $n$ , чтобы  $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} \geq a$ . При  $a=3$  должно получиться  $n=11$ , при  $a=-5$ ,  $n=1$ , при  $a=10$ ,  $n=12367$ .

10. Дано натуральное число  $N$ . Вычислить:  

$$\frac{1}{\sin 1} + \frac{1}{\sin 1 + \sin 2} + \dots + \frac{1}{\sin 1 + \sin 2 + \dots + \sin N}$$

### Задание 3

1. Пусть  $n$  - заданное натуральное число. Введите с клавиатуры  $n$  чисел и определите их среднее арифметическое.
2. Напишите программу, которая запрашивает целые числа с клавиатуры и умножает их до тех пор, пока не будет введено число 0.
3. Написать программу возведения числа  $a$  в степень  $n$ .
4. Найти все меньшие 100 числа-палиндромы, которые при возведении в квадрат также дают палиндромы (палиндромом называют число, которое одинаково читается слева направо и справа налево. Например: 121; 1331; 42524).
5. Дано натуральное число  $n$ . Определить, равны ли первая и последняя цифра.
6. Найти, сколько раз встречается данная цифра в данном целом положительном числе.
7. Среди двузначных чисел найти те, сумма квадратов цифр которых делится на 13.
8. Определить количество натуральных трехзначных чисел, сумма цифр которых равна  $n$ . Значение  $n$  ввести с клавиатуры.
9. Написать программу, которая вычисляет сумму первых  $n$  целых положительных четных чисел. Количество суммируемых чисел должно вводиться во время работы.
10. Сосчитайте, сколько шестизначных чисел имеют одинаковые суммы трех первых и трех последних цифр ("счастливые" числа).

### Задание 4

1. Составить программу для вычисления выражения

$$y = \frac{1}{x^2 + \frac{1}{x^2 + \frac{2}{x^2 + \frac{3}{x^2 + \frac{4}{x^2 + \frac{5}{\dots x^2 + \frac{15}{x^2}}}}}}}$$

при заданном  $x \neq 0$ .

2. Для данного  $n$  составить программу вычисления значения выражения

$$\sqrt{7 + \sqrt{14 + \sqrt{21 + \dots + \sqrt{7n}}}}$$

3. Составить программу вычисления значения выражения  $\sum_{k=1}^n \frac{2x^k}{(k+1)!}$  при заданных  $n$  и  $x$ .
4. Составить программу для вычисления выражения  $y = \frac{1}{1 + \frac{1}{2 + \frac{1}{3 + \frac{1}{4 + \frac{1}{\dots 10 + \frac{1}{11}}}}}}$ .
5. Вычислите  $\sqrt{3 + \sqrt{3 + \sqrt{3 + \dots + \sqrt{3}}}}$  - всего  $n$  корней.
6. Составить программу вычисления значения выражения  $\prod_{x=1}^5 \frac{x^3 - 3.5}{x - |\operatorname{tg} x|}$  при заданных  $x$ .
7. Для данного  $n$  составить программу вычисления значения выражения  $\sqrt{4 + \sqrt{8 + \sqrt{12 + \dots + \sqrt{4n}}}}$ .
8. Для данного  $n$  составить программу вычисления значения выражения  $\sum_{k=0}^n \frac{(-1)^k (k+1)}{k!}$ .
9. Составить программу для вычисления выражения  $y = \frac{x+1}{x + \frac{x+2}{x^2 + \frac{x^2+4}{x^3 + \frac{x^3+8}{x^8 + \dots \frac{256}{x^9}}}}}$  при заданном  $x \neq 0$ .
10. Для данных  $n$  и  $m$  составить программу вычисления значения выражения:  $\prod_{x=m}^n \frac{1-x}{1+x^2} (m < n)$ .

### Лабораторная работа № 3

*Знакомство с компонентами. Процедуры и функции модуля "MATCH"*

**Цель:** рассмотреть компоненты палитры компонент Delphi. Изучить процедуры и функции модуля Match.

#### **Теоретическая часть.**

Компоненты представляют собой элементы, из которых конструируется видимое изображение, создаваемое работающей программой. Заметим, что существует значительное количество компонентов, которые не создают видимого изображения, но которые, тем не менее, играют важную роль в тех или иных случаях. Правильнее думать о компонентах как о заранее

приготовленных для вас фрагментах программы, которые можно вставлять, если в этом есть необходимость, в разрабатываемую программу.

Библиотеки компонентов для разных версий Delphi строятся по принципу расширения: в первой версии было около 70 компонентов, в то время как в состав Delphi 7 входит более 300 компонентов. Мы рассмотрим только основные компоненты, которые будут нам необходимы для дальнейшего изучения среды программирования Borland Delphi.

## **Практическая часть.**

### **СТРАНИЦА STANDARD**

На странице Standard палитры компонентов сосредоточены стандартные для Windows интерфейсные элементы, без которых не обходится практически ни одна программа.

Frame - рама. Наравне с формой служит контейнером для размещения других компонентов. В отличие от формы может размещаться в палитре компонентов, создавая заготовки компонентов. Впервые введен в версию Delphi 5.

MainMenu - главное меню программы. Компонент способен создавать и обслуживать сложные иерархические меню.

PopupMenu - вспомогательное или локальное меню. Обычно это меню появляется в отдельном окне после нажатия правой кнопки мыши.

Label - метка. Этот компонент используется для размещения в окне не очень длинных однострочных надписей.

Edit - строка ввода. Предназначена для ввода, отображения или редактирования одной текстовой строки. Memo - многострочный текстовый редактор. Используется для ввода и/или отображения многострочного текста.

Button - командная кнопка. Обработчик события OnClick этого компонента обычно используется для реализации некоторой команды.

CheckBox - независимый переключатель. Щелчок мышью на этом компоненте в работающей программе изменяет его логическое свойство Checked.

RadioButton - зависимый переключатель. Обычно объединяется как минимум еще с одним таким же компонентом в группу. Щелчок по переключателю приводит к автоматическому освобождению ранее выбранного переключателя в той же группе.

ListBox - список выбора. Содержит список предлагаемых вариантов (опций) и дает возможность проконтролировать текущий выбор.

ComboBox - комбинированный список выбора. Представляет собой комбинацию списка выбора и текстового редактора.

ScrollBar - полоса управления. Представляет собой вертикальную или горизонтальную полосу, напоминающую полосы прокрутки по бокам Windows-окна.

GroupBox - группа элементов. Этот компонент используется для группировки нескольких связанных по смыслу компонентов.

RadioGroup - группа зависимых переключателей. Содержит специальные свойства для обслуживания нескольких связанных зависимых переключателей.

Panel - панель. Этот компонент, как и GroupBox, служит для объединения нескольких компонентов. Содержит внутреннюю и внешнюю кромки, что позволяет создать эффекты “вдавленности” и “выпуклости”.

ActionList - список действий. Служит для централизованной реакции программы на действия пользователя, связанные с выбором одного из группы однотипных управляющих элементов, таких как опции меню, пиктографические кнопки и т. п. Впервые введен в версии Delphi 4.

## СТРАНИЦА ADDITIONAL

В страницу Additional помещены 18 дополнительных компонентов, с помощью которых можно разнообразить вид диалоговых окон.

BitBtn - командная кнопка с надписью и пиктограммой.

SpeedButton - пиктографическая кнопка. Обычно используется для быстрого доступа к тем или иным опциям главного меню.

MaskEdit - специальный текстовый редактор. Способен фильтровать вводимый текст, например, для правильного ввода даты.

StringGrid - таблица строк. Этот компонент обладает мощными возможностями для представления текстовой информации в табличном виде.

DrawGrid - произвольная таблица. В отличие от StringGrid ячейки этого компонента могут содержать произвольную информацию, в том числе и рисунки.

Image - рисунок. Этот компонент предназначен для отображения рисунков, в том числе пиктограмм и метафайлов.

Shape - фигура. С помощью этого компонента вы можете вставить в окно правильную геометрическую фигуру - прямоугольник, эллипс, окружность и т. п.

Bevel - кромка. Служит для выделения отдельных частей окна трехмерными рамками или полосами.

ScrollBar - панель с полосами прокрутки. В отличие от компонента Panel автоматически вставляет полосы прокрутки, если размещенные в нем компоненты отсекаются его границами.

CheckListBox - список множественного выбора. Отличается от стандартного компонента ListBox наличием рядом с каждой опцией независимого переключателя типа CheckBox, облегчающего выбор сразу нескольких опций. Впервые введен в версии 3.

Splitter - граница. Этот компонент размещается на форме между двумя другими видимыми компонентами и дает возможность пользователю во время прогона программы перемещать границу, отделяющую компоненты друг от друга. Впервые введен в версии 3.

StaticText - статический текст. Отличается от стандартного компонента Label наличием собственного windows-окна, что позволяет обводить текст

рамкой или выделять его в виде “вдавленной” части формы. Впервые введен в версии 3.

`ControiBar` - полоса управления. Служит контейнером для “причаливаемых” компонентов в технологии `Drag&Dock`. Впервые введен в версии 4.

`ApplicationEvents` - получатель события. Если этот компонент помещен на форму, он будет получать все предназначенные для программы сообщения `Windows` (без этого компонента сообщения принимает глобальный объект-программа `Application`). Впервые введен в версии 5.

`ValueListEditor` - редактор строк, содержащих пары имя = значение. Пары такого типа широко используются в `Windows`, например, в файлах инициации, в системном реестре и т. п. Впервые введен в версии 6.

`LabeledEdit` - комбинация однострочного редактора и метки.

`ColorBox` - специальный вариант `ComboBox` для выбора одного из системных цветов. Впервые введен в версии 6.

`Chart` - диаграмма. Этот компонент облегчает создание специальных панелей для графического представления данных.

`ActionManager` - менеджер действий. Совместно с тремя следующими компонентами обеспечивает создание приложений, интерфейс которых (главное меню и инструментальные кнопки) может настраиваться пользователем. Впервые введен в версии 6.

`ActionMainMenuBar` - полоса меню, опции которого создаются с помощью компонента `ActionManager`. Впервые введен в версии 6.

`ActionToolBar` - полоса для размещения пиктографических кнопок, создаваемых с помощью компонента `ActionManager`.

`CustomizeDig` - диалог настройки. С помощью этого компонента пользователь может по своему вкусу настроить интерфейс работающей программы. Впервые введен в версии 6.

## **СТРАНИЦА WIN32**

Страница `Win32` содержит интерфейсные элементы для 32-разрядных операционных систем `Windows 95/98/NT/2000`

`TabControl` - набор закладок. Каждая закладка представляет собой прямоугольное поле с надписью и/или рисунком. Выбор той или иной закладки распознается программой и используется для управления содержимым окна компонента.

`PageControl` - набор панелей с закладками. Каждая панель может содержать свой набор интерфейсных элементов и выбирается щелчком по связанной с ней закладке.

`ImageList` - набор рисунков. Представляет собой хранилище для нескольких рисунков одинакового размера.

`RichEdit` - многострочный редактор форматированного текста. В отличие от компонента `Memo` страницы `Standard` текст в компоненте `RichEdit` подчиняется правилам Расширенного Текстового Формата (`RTF` - `Rich Text`

Format) и может изменять такие свои характеристики, как шрифт, цвет, выравнивание и т. д.

TrackBar - регулятор. Используется для управления значениями некоторых величин в программах. Например, с его помощью удобно изменять громкость звучания в мультимедийных программах.

ProgressBar - индикатор процесса. С помощью этого компонента можно отображать ход выполнения достаточно длительного по времени процесса, например, процесса переноса данных на дискету.

UpDown - цифровой регулятор. Две кнопки этого компонента предназначены для увеличения (верхняя) или уменьшения (нижняя) связанной с компонентом числовой величины.

HotKey - управляющая клавиша. Компонент используется для ввода управляющих клавиш, таких как F1, Alt+A, Ctrl+Shift+1 и т. п.

Animate - мультипликатор. Предназначен для отображения последовательно сменяющихся друг друга кадров движущихся изображений (видеоклипов). Компонент не может сопровождать видеоклип звуком. Впервые введен в версии 3.

DateTimePicker - селектор времени/даты. Этот компонент предназначен для ввода и отображения даты или времени. Впервые введен в версии 3.

TreeView - дерево выбора. Представляет собой совокупность связанных в древовидную структуру пиктограмм. Обычно используется для просмотра структуры каталогов (папок) и других подобных элементов, связанных иерархическими отношениями.

ListView - панель пиктограмм. Организует просмотр нескольких пиктограмм и выбор нужной. Этот компонент способен располагать пиктограммы в горизонтальных или вертикальных рядах и показывать их в крупном или мелком масштабе.

HeaderControl- управляющий заголовок. Представляет собой горизонтальную или вертикальную полосу, разделенную на ряд смежных секций с надписями. Размеры секций можно менять мышью на этапе работы программы. Обычно используется для изменения размеров столбцов или строк в разного рода таблицах.

StatusBar - панель статуса. Предназначена для размещения разного рода служебной информации в окнах редактирования. Посмотрите на нижнюю часть рамки окна кода Delphi или текстового редактора Word, и вы увидите этот компонент в действии.

ToolBar - инструментальная панель. Этот компонент служит контейнером для командных кнопок BitBtn и способен автоматически изменять их размеры и положение при удалении кнопок или при добавлении новых. Впервые введен в версии 3.

CoolBar - инструментальная панель. В отличие от ToolBar используется как контейнер для размещения стандартных интерфейсных компонентов Windows, таких как Edit, ListBox, ComdoBox и т. д. Впервые введен в версии 3.

PageScroller - прокручиваемая панель. Служит для размещения узких инструментальных панелей. При необходимости автоматически создает по краям панели стрелки прокрутки. Впервые введен в версии 4.

ComboBoxEx - компонент в функциональном отношении подобен comboBox (страница standard), но может отображать в выпадающем списке небольшие изображения. Впервые введен в версии 6.

## **СТРАНИЦА SYSTEM**

На этой странице представлены компоненты, которые имеют различное функциональное назначение, в том числе компоненты, поддерживающие стандартные для Windows технологии межпрограммного обмена данными OLE (Object Linking and Embedding -связывание и внедрение объектов) и DDE (Dynamic Data Exchange -динамический обмен данными). Технология DDE в современных программах полностью вытеснена гораздо более мощной технологией COM и поэтому соответствующие им компоненты в книге не рассматриваются.

Timer - таймер. Этот компонент служит для отсчета интервалов реального времени.

PaintBox - окно для рисования. Создает прямоугольную область, предназначенную для прорисовки графических изображений.

MediaPlayer - мультимедийный проигрыватель. С помощью этого компонента можно управлять различными мультимедийными устройствами.

OleContainer - OLE-контейнер. Служит приемником связываемых или внедряемых объектов.

Компоненты этой страницы имеются во всех предыдущих версиях Delphi.

## **СТРАНИЦА DIALOGS**

Компоненты страницы Dialogs реализуют стандартные для Windows диалоговые окна.

OpenDialog - открыть. Реализует стандартное диалоговое окно “Открыть файл”.

SaveDialog - сохранить. Реализует стандартное диалоговое окно “Сохранить файл”.

OpenPictureDialog - открыть рисунок. Реализует специальное окно выбора графических файлов с возможностью предварительного просмотра рисунков.

SavePictureDialog - сохранить рисунок. Реализует специальное окно сохранения графических файлов с возможностью предварительного просмотра рисунков.

FontDialog - шрифт. Реализует стандартное диалоговое окно выбора шрифта.

ColorDialog - цвет. Реализует стандартное диалоговое окно выбора цвета.

PrintDialog - печать. Реализует стандартное диалоговое окно выбора параметров для печати документа.

PrinterSetupDialog - настройка принтера. Реализует стандартное диалоговое окно для настройки печатающего устройства.

FindDialog - поиск. Реализует стандартное диалоговое окно поиска текстового фрагмента.

ReplaceDialog - замена. Реализует стандартное диалоговое окно поиска и замены текстового фрагмента.

Компоненты OpenPictureDialog И SavePictureDialog введены в версии 3, остальные имеются в предыдущих версиях. Разумеется, интерфейс окон для Windows 16 (версия 1) отличается от интерфейса Windows 32.

## **СТРАНИЦА SAMPLES**

Эта страница содержит компоненты разного назначения.

Gauge - индикатор состояния. Подобен компоненту ProgressBar (страница Win32), но отличается большим разнообразием форм.

ColorGrid - таблица цветов. Этот компонент предназначен для выбора основного и фоновых цветов из 16-цветной палитры.

SpinButton - двойная кнопка. Дает удобное средство управления некоторой числовой величиной.

SpinEdit - редактор числа. Обеспечивает отображение и редактирование целого числа с возможностью его изменения с помощью двойной кнопки.

DirectoryOutLine - список каталогов. Отображает в иерархическом виде структуру каталогов дискового накопителя.

Calendar - календарь. Предназначен для показа и выбора дня в месяце.

## **КОМПОНЕНТЫ ДЛЯ РАБОТЫ С БАЗАМИ ДАННЫХ**

В Delphi развиты средства построения приложений, рассчитанных на работу с электронными архивами (базами данных). Причем Delphi 6 предоставляет программисту возможность выбора способа доступа к данным: это может быть стандартный для ранних версий Delphi доступ с помощью машины баз данных BDE (Borland Database Engine), усиленно развиваемая Microsoft технология ADO ActiveX Data Objects), прямое управление сервером InterBase с помощью технологии IBExpress, наконец, технология dbExpress для непосредственного обращения к промышленным серверам MySQL, DB2, Oracle и некоторым другим.

## **Страница Data Access**

В отличие от предыдущих версий на этой странице собраны компоненты, которые не зависят от используемого доступа к базе данных (большинство компонентов с этой страницы предыдущих версий перекочевали на страницу bde). Они в основном используются в так

называемых трехзвенных БД (с сервером приложений). Часть компонентов известны по другим версиям Delphi, часть появилась в версии 6.

### **Страница Data Controls**

15 компонентов этой страницы предназначены для визуализации данных, их ввода и редактирования. Многие компоненты этой страницы введены еще в версии 1.

### **Страница dbExpress**

7 компонентов, представленных на этой странице, поддерживают технологию dbExpress прямого доступа к некоторым промышленным серверам баз данных. Все компоненты страницы впервые введены в версии 6.

### **Страница DataSnap**

На этой странице сосредоточены компоненты, реализующие взаимодействие машин в локальной сети или Интернет в типичном для БД случае, когда клиент работает с удаленными данными. Часть компонентов известны по другим версиям Delphi, часть появилась в версии 6.

### **Страница BDE**

Здесь представлены компоненты, поддерживающие доступ к данным с помощью BDE - Table, Query, StoredProc И Т. П. Механизм BDE в равной степени успешно работает как с файл-серверными, так и клиент-серверными БД. Компоненты этой страницы есть во всех версиях Delphi.

### **Страница ADO**

Компоненты этой страницы в функциональном отношении во многом подобны компонентам страницы BDE, но поддерживают доступ к данным с помощью технологии ADO (ADOTable, ADOQuery, ADOSToredProc и т. д.). Все компоненты страницы впервые введены в версии 5.

### **Страница InterBase**

“Родной” для Delphi сервер баз данных InterBase (производитель - InterBase Software Corporation - является дочерним предприятием Borland) имеет непосредственную поддержку в виде компонентов этой страницы. В них используется технология IBExpress, позволяющая отказаться от BDE, ADO или иных подобных механизмов доступа к данным. Все компоненты страницы впервые введены в версии 5.

### **Страница Decision Cube**

На этой странице представлены компоненты для систем принятия решений на основании анализа многомерных наборов данных. Компоненты этой страницы впервые введены в версии 3.

## **QReport**

Около 30 компонентов страницы предназначены для упрощения создания отчетов по материалам, хранящимся в БД. Большинство компонентов страницы впервые введено в версии 2.

Так же существуют страницы с компонентами для доступа в Интернет и доступа к серверам автоматизации.

## **ПРОЦЕДУРЫ И ФУНКЦИИ МОДУЛЯ MATCH**

### **Тригонометрические подпрограммы**

function ArcCos(X: Extended): Extended – арккосинус;

function ArcSin(X: Extended): Extended – арксинус;

function ArcTan2(Y, X: Extended): Extended - вычисляет арктангенс Y/X и возвращает угол в правильном квадранте (функция ArcTan модуля System не учитывает квадрант);

function Cotan(X: Extended): Extended – котангенс;

function Hypot (X, Y: Extended): Extended - корень квадратный из (X\*\* 2 + Y\*\* 2)-гипотенуза прямоугольного треугольника по двум катетам;

procedure SinCos (Theta: Extended; va: "Sin, Cos: Extended) - возвращает одновременно синус и косинус угла Theta (почти в 2 раза быстрее, чем раздельное получение синуса и косинуса);

function Tan(X: Extended): Extended – тангенс;

### **Функции преобразования углов**

function CycleToRad(Cycles: Extended) : Extended - Radians := Cycles \* 2PI;

function DegToRad(Degrees: Extended) Extended - Radians := Degrees \* PI / 18;

function GradToRad(Grads: Extended): Extended - Radians := Grads \* PI / 200;

function RadToDeg(Radians: Extended) Extended - Degrees := Radians \* 180 / PI;

function RadToGrad(Radians: Extended) : Extended - Grads := Radians \* 200 / PI;

function RadToCycle(Radians: Extended) : Extended - Cycles := Radians / 2PI;

### **Подпрограммы разного назначения**

function Ceil(X: Extended): Integer - ближайшее меньшее целое;

function Floor (X: Extended): Integer - ближайшее большее целое;

procedure Frexp(X: Extended; var Mantissa: Extended; var Exponent: Integer) - возвращает мантиссу и степень вещественного числа;

function Ldexp(X: Extended; P: Integer) : Extended - возвращает X\*P\*P;

### **Статические программы**

function Max (A, B: Integer): Integer; overloads - возвращает максимальное из двух чисел;

function Poly(X: Extended; const Coefficients: array of Double): Extended - значение полинома  $A \cdot X^n + B \cdot X^{n-1} + \dots + Z$ . Коэффициенты задаются в порядке возрастания степени;

### **Экспоненциальные функции**

function IntPower(Base: Extended; Exponent: Integer) : Extended - возведение Base в целочисленную степень Exponent;

function Power(Base, Exponent: Extended) : Extended - возведение Base в вещественную степень Exponent;

### **Контрольные вопросы**

1. Поясните назначение визуальных компонент палитры компонент.
2. Поясните назначение процедур и функций модуля MATN.

### **Задание**

Составить и выполнить программу по обработке символьных данных. В программе должны быть использованы 3-5 визуальных компонентов (выбираются самостоятельно студентом). Поработать со свойствами компонент и формы.

**Задание 1.** Текст вводится с клавиатуры в символьную переменную. Исходный текст и результаты распечатать.

1. В тексте заменить все вхождения одного символа на другой.
2. Ввести строку и затем еще слово. Исключить из строки все вхождения этого слова.
3. Определить количество пробелов в заданной строке.
4. В данной строке подсчитать количество вхождений каждой буквы.
5. Найти количество слов в строке.
6. Ввести строку и символ. Вывести строку, полученную из исходной строки удалением всех вхождений этого символа.
7. Заменить во введенной строке первую букву каждого слова на заглавную.
8. Заданы две строки. Найти самую длинную их общую (одинаковую) часть.
9. Определить количество вхождений заданного символа в заданную строку.
10. Удалить из строки повторные вхождения каждого символа.

### **Задание 2**

1. Дана символьная строка. Русские буквы *a, e, o, э* в ней нужно удвоить, а между словами оставить только по одному пробелу.
2. Дана символьная строка. Если какой-то символ встречается в ней более одного раза, первое вхождение этого символа оставить без изменения, второе – заменить цифрой «2», третье – «3» и т.д.
3. Дана символьная строка, содержащая русские слова, записанные строчными буквами, разделенные пробелами. Заменить первые буквы слов на прописные, а между ними оставить по одному пробелу.
4. Дана последовательность символов (строка). Если какой-то символ в ней встречается более одного раза, оставить только первое вхождение.

5. Дана символьная строка, содержащая русские буквы, цифры и пробелы. Написать строку задан наперед, удалив из нее все цифры и пробелы.
6. Дана символьная строка. Определить, содержит ли строка числа, если да, то вывести на экран только четные.
7. Дана символьная строка, состоящая из строчных букв русского и латинского алфавита без пробелов. Гласные латинские буквы в ней нужно заменить на соответствующие прописные буквы, а каждые пять символов разделить пробелами.
8. Дана символьная строка. Вывести на экран только те символы, которые встречаются в ней более трех раз, для каждого из этих символов указать точное количество повторений.
9. Дана символьная строка. Удалить из нее все символы, не являющиеся заглавными буквами русского или латинского алфавита.
10. Дана символьная строка, содержащая два предложения, каждое из которых заканчивается точкой. Поменять их местами, сохранив порядок слов в предложениях.

#### **Лабораторная работа № 4**

##### *Создание меню*

**Цель:** изучить визуальные компоненты для создания основного и дополнительного меню. Рассмотреть свойства компонент для создания меню.

##### **Теоретическая часть**

Меню, как элемент интерфейса, присутствовали уже в самых первых программах, работавших еще в среде DOS. За прошедшие годы роль меню не уменьшилась — любое современное приложение имеет меню. Причем все это время меню совершенствовалось. Такая популярность этого элемента интерфейса приложений объясняется просто — меню обеспечивает стандартный и удобный доступ к функциям приложения и прекрасно структурирует их в однородные группы.

В Палитру компонентов Delphi включены два компонента меню. Компонент TMainMenu обеспечивает создание полосы главного меню приложения. Компонент TPopupMenu представляет собой всплывающее меню, которое может быть придано любому визуальному компоненту.

Меню не являются элементами управления (Controls) и становятся визуальными только при выполнении приложения. Во время разработки меню конструируются при помощи специального Редактора меню. Он открывается при щелчке на кнопке свойства items компонента меню в Инспекторе объектов или при двойном щелчке на компоненте. Он обеспечивает отображение всех элементов меню, представленных в свойстве items.

Редактор меню активно взаимодействует с Инспектором объектов. Для каждого выбранного в Редакторе элемента в Инспекторе объектов появляются значения его свойств. Перемещаясь по элементам, разработчик может редактировать их свойства.

Для того чтобы добавить новый элемент в конец меню, достаточно выбрать пустой элемент, который всегда имеется в конце списка, и заполнить его свойства.

Для выбранного элемента меню в Инспекторе объектов активизируется свойство `caption`, что позволяет быстро вводить текст команды меню и переходить к следующему элементу. Если в тексте команды встречается символ амперсанта (&), то следующий за ним символ автоматически становится горячей клавишей и подчеркивается.

Если в качестве текста элемента меню введен символ "-", то этот элемент меню отображается как разделитель.

Компонент `TMainMenu` инкапсулирует главное меню.

Компонент `TPopupMenu` позволяет создавать всплывающие (контекстные) меню.

Пользовательский интерфейс современных приложений весьма многообразен, и зачастую один и тот же результат можно получить разными способами. Можно решить проблему "в лоб" и повторить один и тот же код два, три раза и так далее. Недостатки такого подхода не обязательно комментировать. Можно воспользоваться Инспектором объектов и назначить пункту меню тот же обработчик события, что и кнопке, благо событие `onClick` везде одинаковое.

### **Практическая часть**

Наконец, можно воспользоваться компонентом `TActionList` и разом решить все проблемы. Как следует из названия, это — централизованное хранилище, где воздействия со стороны пользователя связываются с реакциями на них.

Для начала работы следует поместить в форму компонент `TActionList`. Он находится в Палитре компонентов на первой странице `Standard`. После этого следует запустить редактор списка действий двойным щелчком мышью на компоненте или с помощью контекстного меню.

Теперь можно начинать добавление действий. Для программиста предусмотрен набор типовых, наиболее часто встречающихся действий. К ним относятся операции для работы с Буфером обмена (`Copy/Cut/Paste`), упорядочения окон (`Tile/Cascade` и т. п.), а также типовые операции по работе с наборами данных (перемещение, вставка, замена и прочие).

### **Контрольные вопросы**

1. Назначение, функции и свойства компонента `MainMenu`.
2. Назначение, функции и свойства компонента `PopupMenu`.
3. Назначение, функции и свойства компонента `ActionList`.

### **Задание**

Создать на форме основное и дополнительное меню для вызова функций, реализованных в предыдущих лабораторных работах.

## Лабораторная работа № 5 Использование графики

**Цель:** изучить базовые возможности Borland Delphi по использованию графики.

### **Теоретическая часть.**

#### **Графические инструменты Delphi**

Разработчики Delphi уделили большое внимание возможностям работы с деловой графикой: простота и удобство ее использования напрямую сказываются на простоте и удобстве созданных приложений. Вместо дебрей графического интерфейса Windows разработчик получил несколько инструментов столь понятных, столь же и мощных.

В стандартном графическом интерфейсе Windows (GDI) основой для рисования служит дескриптор контекста устройства hDC и связанные с ним шрифт, перо и кисть. В состав VCL входят объектно-ориентированные надстройки над последними, назначением которых является удобный доступ к свойствам инструментов и прозрачная для пользователя обработка всех их изменений.

### **Практическая часть.**

#### **Класс TFont**

Класс инкапсулирует шрифт Windows. В Delphi допускаются только горизонтально расположенные шрифты. В конструкторе объекта по умолчанию принимается шрифт System цвета clWindowText и размером 10 пунктов.

#### **Класс TPen**

Этот класс инкапсулирует свойства пера GDI Windows. В конструкторе по умолчанию создается непрерывное (psolid) черное перо шириной в один пиксел.

#### **Класс TBrush**

Этот класс инкапсулирует свойства кисти — инструмента для заливки областей. Когда создается экземпляр этого класса, первоначально создается белая сплошная (style-bssolid) кисть.

#### **Класс TCanvas**

Этот класс — сердцевина графической подсистемы Delphi. Он объединяет в себе и "холст" (контекст конкретного устройства GDI) и "рабочие инструменты" (перо, кисть, шрифт), и даже "подмастерьев" (набор функций по рисованию типовых геометрических фигур). Будем называть его канвой.

Тем самым мы рассмотрели только основные классы, но на самом деле их значительно больше.

## Контрольные вопросы

1. Какие графические инструменты Delphi Вы знаете?
2. Каковы базовые возможности Borland Delphi по использованию графики?

## Задание

Создать программу рисования сложного графического объекта. В программе должны меняться как минимум три параметра объекта (например, скорость движения, направление движения, цвет объекта).

### Лабораторная работа № 6

#### *Создание встроенной справочной системы*

**Цель:** научиться создавать встроенную справочную систему для приложений

#### **Теоретическая часть**

Для реализации процесса создания справки вам понадобятся как минимум две утилиты: текстовый RTF-редактор и компилятор Help-файлов. В качестве текстового редактора обычно используется Microsoft Word (далее просто Word), поддерживающий расширенный текстовый формат RTF (стандартный для Windows 32 текстовый редактор WordPad также поддерживает формат RTF, но в нем нет удобных средств вставки специальных управляющих символов). В ранних версиях Delphi в качестве компилятора Help-файлов поставлялся стандартный компилятор hcsi.exe. С последними версиями 4 и 5 поставляется более удобная утилита Microsoft Help Workshop (файл hew.exe из каталога help | tools каталога размещения Delphi).

#### **Практическая часть.**

##### **ЭТАПЫ РАЗРАБОТКИ**

Разработка Help-службы требует решения следующих основных задач.

*Планирование системы справок.* На этом этапе составляется перечень разделов справочной службы и необходимых перекрестных ссылок.

*Создание текстовых файлов,* содержащих описания справочных разделов. Текстовые файлы готовятся с помощью любого текстового редактора, поддерживающего расширенный текстовый формат RTF. В них включаются специальные управляющие символы для создания перекрестных ссылок и подключения растровых изображений.

*Разработка проектного файла,* содержащего специальные команды для Help-компилятора. Проектный файл описывает структуру справочной службы в целом, в нем каждому разделу присваивается уникальный целочисленный идентификатор.

*Разработка файла содержания.* Содержание активизируется при запуске HLP-файла, а также после щелчка по кнопке содержание в окне справочной службы.

*Компиляция Help-файлов.*

*Тестирование и отладка Help-службы.*

*Связь программы с разделами Help-службы.*

## **ПЛАНИРОВАНИЕ СИСТЕМЫ СПРАВОК**

На этапе планирования необходимо составить перечень разделов справочной службы и нужных перекрестных ссылок. Структура разделов и количество перекрестных ссылок зависит от сложности программы, для которой создается Help-служба.

### **Полезно учесть следующие рекомендации.**

Любой раздел может содержать список подчиненных или связанных разделов, т. е. разрешается произвольная иерархия справочной службы.

Полезно структурировать разделы по роду предоставляемой ими информации. Например, предусмотреть разделы для начинающего пользователя с пошаговыми инструкциями, выделить примеры использования программы, предоставить информацию опытным пользователям с точным и детальным описанием каждой программной функции, а также информацию о “продвинутых” возможностях программы и т. п.

Каждый раздел должен, по возможности, отображаться в одном распахнутом окне. Избегайте слишком длинных пояснений, лучше разбейте такой комментарий на несколько разделов. Учтите, что согласно исследованиям лишь менее 30% людей могут читать текст на экране так же легко и быстро, как на бумаге.

Структурируйте текст, вставляйте в него больше рисунков, выделяйте важную информацию шрифтами и цветом. Помните, что монотонный текст (как и чрезмерно “раскрашенный”) резко увеличивает утомляемость пользователя и затрудняет усвоение информации.

## СОЗДАНИЕ ТЕКСТОВЫХ ФАЙЛОВ

Вся справочная информация помещается в один или несколько текстовых файлов в формате RTF. Для их создания может использоваться любой текстовый редактор, поддерживающий этот формат.

При создании текстовых файлов учтите следующие особенности их подготовки:

Количество RTF-файлов может быть произвольным. Вы можете поместить всю справочную информацию в один файл или предусмотреть для каждого раздела отдельный файл - все зависит от объема и структуризации Help-службы и ваших личных пристрастий.

В пределах одного файла каждый раздел отделяется от другого служебным символом конца страницы (с помощью Word этот символ вставляется клавишами Ctrl+Enter).

Любой раздел, доступный с помощью перекрестных ссылок или индексных указателей, должен иметь связанный с ним идентификатор - уникальную текстовую строку.

Раздел может иметь название и связанный с ним список ключевых слов.

*Сноска “#”* Перечеркнутый или дважды подчеркнутый текст *Подчеркнутый текст* - Указывает идентификатор раздела, определяет отображение связанного с текстом раздела (перекрестной ссылки) в стандартном справочном окне. Определяет отображение раздела перекрестной ссылки в окне пояснений (“всплывающем” окне);

*Скрытый текст* - Определяет идентификатор раздела, связанного с перекрестной ссылкой;

*Сноска “\$”* - Задаёт название раздела;

*Сноска “K”* - Указывает список ключевых слов для поиска раздела;

*Сноска “+”* - Задаёт порядковый номер раздела в списке просмотра связанных разделов.

*Сноска “^”* - Определяет условие компиляции раздела;

*Сноска “>”* - Определяет тип дополнительного окна, в котором будет отображаться раздел.

*Сноска “!”* - Указывает макрокоманду, которая будет выполняться при открытии окна с разделом.

### **Задание идентифицирующей строки и организация перекрестных ссылок**

Для задания перекрестных ссылок, реализующих скачок от одного раздела к другому, разделы помечаются уникальными идентифицирующими строками (идентификаторами). Только помеченные идентификаторами разделы можно просматривать в рамках гипертекстовой системы (непомеченные разделы могут быть доступны для просмотра по ключевым словам и в порядке просмотра связанных разделов).

Идентифицирующая строка может содержать любые символы, кроме #, @, !, \*, =, >, % и пробелов. Разница в высоте латинских букв (но не кириллицы!) игнорируется. Длина строки - до 255 символов.

В качестве идентификаторов имеет смысл использовать текст заголовка раздела, в котором пробелы заменены символами подчеркивания, в этом случае вам не придется вспоминать идентификатор при ссылке на него.

Идентификатор задается с помощью сноски “#” в самом начале раздела.

Перекрестная ссылка представляет собой выделенную цветом часть текста, щелчок мышью по которой приводит к смене раздела.

Для кодирования ссылки она выделяется в тексте перечеркнутым или дважды подчеркнутым шрифтом, и сразу за ней без каких либо пробелов указывается идентификатор темы в виде скрытого текста.

Раздел «Перекрестные ссылки» связан с идентификатором hypertext, который задан в виде текста сноски “#”, предшествующей первому символу заголовка (для задания сноски выберите в главном меню Word'a Вставка | Сноска, в окне Сноска выберите Другая и введите символ “#” в строке 1, 2, 3).

В тексте раздела указаны две перекрестные ссылки в виде слов ссылка и гипертекст. Эти слова выделены соответственно дважды подчеркнутым и перечеркнутым шрифтом, но в окне справочной службы они будут выглядеть одинаково: выделены зеленым цветом и подчеркнуты одной линией. Сразу за каждой ссылкой следует идентификатор соответствующего раздела.

Поскольку идентификатор выделяется скрытым текстом в обычном режиме, он скрыт и становится виден только в режиме отображения служебных символов после щелчка по кнопке V на инструментальной панели Word.

Для задания скрытого текста в редакторе Word необходимо выделить текст и применить к нему команду Hidden. Для этого можно воспользоваться опцией Формат | Шрифт и в разделе Эффекты щелкнуть по опции скрытый.

### **Задание названия раздела**

Название раздела используется в Help-службе следующим образом:

- оно появляется после активизации опции «Закладка» в главном меню Help-службы;
- оно указывается в списке разделов диалоговых окон, связанных с кнопками «Поиск» и «Хронология» инструментальной панели справочного окна.

Название раздела задается с помощью сноски “\$”, которая должна предшествовать первому символу текста раздела. Название раздела записывается в тексте сноски и отделяется от символа “\$” одним пробелом. В тексте названия можно указывать любые символы, в том числе и пробелы. Максимальная длина названия - 255 символов. Обычно название раздела совпадает с его заголовком.

### **Определение ключевых слов**

Help-служба позволяет искать разделы по связанным с ними ключевым словам. В диалоговом окне Поиск предусмотрены строка для ввода ключевых слов и возможность выбора их из списка. Для любого раздела можно назначить сколько угодно ключевых слов и, наоборот, с любым ключевым словом можно связать сколько угодно разделов.

Для определения ключевого слова в начале раздела (до первого символа текста раздела) ставится сноска, помеченная латинской буквой “K” или “k”.

## **РАЗРАБОТКА ПРОЕКТНОГО ФАЙЛА**

Проектный файл служит основным управляющим документом для Help-компилятора. В Windows 32 он создается с помощью утилиты Microsoft Help Workshop (MS HW) и представляет собой ASCII-текстовый файл, содержащий несколько секций. Секция - это фрагмент текста, состоящий из заголовка и одной или нескольких следующих за ним строк (опций) вида

ИМЯ\_ПАРАМЕТРА = ЗНАЧЕНИЕ,  
подобно тому как организованы файлы инициации ini.

Для запуска MS HW следует загрузить файл help | tools | hcw.exe из папки размещения Delphi. Если файл проекта уже был ранее создан, для работы с ним используется опция File | open или соответствующая кнопка инструментальной панели MS HW. Для создания нового проектного файла выберите File | New и в дополнительном диалоговом окне - Help project.

Всю центральную часть окна занимает текст проектного файла, который первоначально состоит из единственной секции (options). В отличие от обычных текстовых редакторов это окно недоступно для клавиатурного ввода: чтобы вставить в него текст или изменить уже введенные секции, используются расположенные справа кнопки.

### **Секция OPTIONS**

Секция [Options] создается или изменяется с помощью кнопки options. Оно содержит общие для всей справочной службы установки и, в частности, определяет умалчиваемую тему, показываемую при открытии HELP-службы, заголовок основного окна, используемый в текстах язык и т. п.

### **Секция FILES**

С помощью этой секции определяются RTF-файлы, содержащие текст справочной системы. Для нормальной работы компилятора необходимо определить хотя бы один файл в этой секции. Замечу, что содержимое этой секции автоматически изменяется при изменении списка RTF-files на странице Files окна options.

### **Секция WINDOWS**

Секция [windows] определяет используемый в справочной службе набор дополнительных окон. Каждое указанное в секции окно характеризуется

своим типом, цветом, размерами, заголовком. Только имена перечисленных в секции окон могут указываться в ссылках и сносках “>” RTF-файлов.

### **Секция MAP**

В этой секции следует определить числовые идентификаторы разделов, которые будут автоматически вызываться кнопками Help и клавишей F1 в работающей программе. Напомню, что все видимые компоненты Delphi имеют специальное свойство HelpContext, в которое можно поместить числовой идентификатор раздела справочной службы. Если соответствующий компонент имеет фокус ввода и пользователь нажимает кнопку F1, автоматически включается справочная система Windows, которая отыскивает в help-файле и показывает в справочном окне соответствующий раздел. Только разделы, указанные в секции [map], могут быть контекстно-доступными из работающей программы. Остальные разделы можно просмотреть после вызова справки с помощью выбора ключевых слов кнопками просмотра связанных разделов или по содержанию справочной службы.

Компиляция и тестирование справочной службы не составляют проблемы: с помощью MS HW создайте (или загрузите) проектный файл и щелкните по кнопке save and compile или по инструментальной кнопке.

Связь с программой реализуется с помощью свойств HelpContext видимых компонентов, в которые следует поместить числовые идентификаторы нужных разделов справочной службы так, как они определены в секции MAP. Кроме того, в свойство Application.HelpFile нужно поместить имя help-файла. Обычно эта связь устанавливается в обработчике события OnCreate главной формы программы. После такой настройки пользователь программы сможет с помощью клавиши F1 получить контекстно-чувствительную справку, т. к. при нажатии F1 автоматически вызывается раздел, числовой идентификатор которого помещен в свойство HelpContext компонента с фокусом ввода. Если HelpContext компонента с фокусом ввода содержит 0, вызывается раздел, указанный в HelpContext его владельца, а если и у того это свойство не определено, используется HelpContext активной формы (если во всей цепочке владельцев, включая активную форму, свойство HelpContext не определено, нажатие F1 игнорируется).

### **Контрольные вопросы**

1. Каковы основные этапы создания файлов справки?
2. С помощью каких программ создаются файлы справки?

### **Задание**

Создать к любой лабораторной работе файл помощи и подключить его к приложению.



