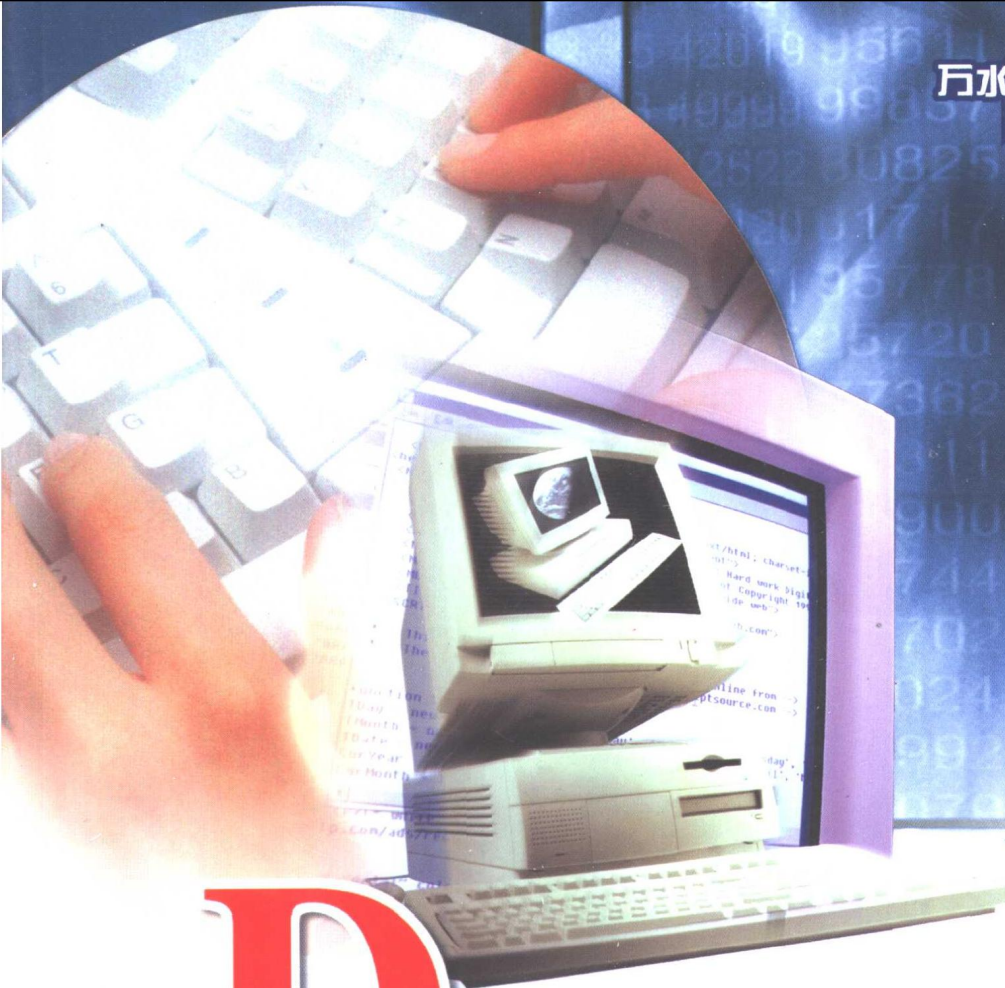


万水创作效果百例丛书



# Delphi 7

## 精彩编程百例

黄卓 刘祯祥 等编著



附光盘



中国水利水电出版社  
www.waterpub.com.cn

万水创作效果百例丛书

# Delphi 7 精彩编程百例

黄 卓 刘祯祥 等编著

中国水利水电出版社

## 内 容 提 要

Delphi 7 是 Inprise (原 Borland) 公司最新推出的面向对象的可视化编程语言, 具有良好的面向对象和控件开发的性能。它提供了大量可重用控件和用户自建模板技术, 极大地提高了应用系统的开发速度, 使其成为目前最优秀的前端开发平台之一。

本书以实例演示的形式, 深入浅出地讲解了使用 Delphi 7 进行程序开发的各种技巧。全书共 100 个实例, 典型而又有针对性。内容分 6 篇, 包括 Delphi 7 编程基础及控件应用、窗体设计方法、文件与系统的操作技巧、图形图像的制作和显示技巧、多媒体技术、数据库编程、网络技术等。全面而系统地介绍了 Delphi 7 程序设计的基本理论与实用编程技术。

本书适合初中级读者学习使用, 也可以作为 Delphi 开发人员的参考书籍。

### 图书在版编目 (CIP) 数据

Delphi 7 精彩编程百例/黄卓等编著. —北京: 中国水利水电出版社, 2003  
(万水创作效果百例丛书)

ISBN 7-5084-0702-4

I. D… II. 黄… III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2003) 第 060129 号

书 名	Delphi 7 精彩编程百例
作 者	黄卓 刘桢祥 等编著
出版、发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@public3.bta.net.cn (万水) sale@waterpub.com.cn
经 售	电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水) 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787×1000 毫米 16 开本 23 印张 498 千字
版 次	2003 年 8 月第一版 2003 年 8 月北京第一次印刷
印 数	0001—5000 册
定 价	36.00 元

凡购买我社图书, 如有缺页、倒页、脱页的, 本社营销中心负责调换

版权所有·侵权必究

# 前 言

Delphi 是 Inprise 公司推出的一个可视化、面向对象的快速应用程序开发工具，以其快速高效的开发方式和强大的编译功能得到软件开发者和程序员的青睐和欢迎，现已成为使用最广泛的强力开发工具。

Delphi 实际上是 Pascal 语言的一种版本，但它与传统的 Pascal 语言有天壤之别。它继承了 Pascal 语言代码结构清晰、可读性好和代码执行效率高等优点，但它又包含了一个便捷的开发设计框架——缺省窗体。Delphi 把 Windows 编程的回调、句柄处理等过程都放在一个不可见的覆盖物下面，忽略了很多烦琐的步骤和响应过程。而其本身的集成开发环境不仅提供了一整套用于设计、编写、测试、调试和发布应用程序的工具软件，同时还包含了大量可以重用的控件和用户自建模板，极大地提高了应用系统的开发速度。

面向对象的程序设计是 Delphi 的另一大特点。它通过给程序中加入扩展语句，把函数“封装”进 Windows 编程所必需的“对象”中，使得复杂的工作条理清晰、编写容易。完全面向对象这一特点使得 Delphi 成为一种触手可及的促进软件重用的开发工具。

Delphi 是工作在 Windows 9x、Windows 2000、Windows NT 或 Windows XP 操作系统上的。使用 Delphi 7 可以编写 Win 32 控制台应用程序和具有图形用户界面的 Win 32 应用程序。和传统的 Windows 应用程序的开发方法相比，Delphi 具有快速和真正可视化的特点。可以说，Delphi 是可视化编程和面向对象开发框架的结合体。

本书的主要形式是实例讲解，全书针对不同层次读者及不同的开发应用方面共编写了 100 个实例。这 100 个实例突出地讲解了 Delphi 7 的开发思想、技术和方法，极具典型性。对于每个实例，编者又将其分成“实例说明”、“技术指南”、“创作步骤”、“程序代码”和“结果演示”5 个部分，分别进行详细说明。同时本书附带有实例光盘，便于读者查看每个实例的详细源代码和控件的具体属性设置。

全书共分为 6 篇，内容广泛、覆盖面广，主要讲解了 Delphi 7 编程基础及控件应用、各种窗体的设计方法、文件与系统的调用和管理、图形图像制作和显示技巧、多媒体技术、数据库设计管理编程以及网络互连技术这 6 个方面的内容，基本包括了 Delphi 程序设计开发的所有方面。

本书清晰明了、深入浅出，每个实例都有详细的制作步骤和精确的图像演示，对源代码的关键部分都有详细的解释说明，十分适合初中级用户学习使用。

参加本书编写工作的除黄卓、李鑫外，还有刘祜祥、李蓉、王明凯、孔玲飞、张远、蔡振华、赵晓明、李明、赵海斌、王鹏、韩天雷、刘建峰、马明宇、刘欣、邹建鹏、范晨毅、罗静、

周鑫华、刘建华、王逸凡、吴玉青、耿云、马纪明、张明、高一波、杜马辰等，在此感谢他们所做的工作，另外还要感谢中国水利水电出版社北京万水公司的编辑同志们，是他们的辛勤工作才使得本书顺利出版。由于笔者水平有限，经验不多，书中难免会有不足和错误，欢迎各位专家读者批评指正，促进大家共同进步。

编者

2003年6月

# 目 录

## 第一篇 Delphi 7 编程基础及控件应用

实例 1	加载图形按钮 .....	2
实例 2	简易计算器 .....	5
实例 3	绘制多种曲线 .....	8
实例 4	动态改变鼠标 .....	12
实例 5	制作工具栏 .....	14
实例 6	查看计算机硬件信息 .....	18
实例 7	DirectoryListBox 与 DriveComboBox 控件 .....	21
实例 8	制作图表 .....	24
实例 9	自定义下拉菜单 .....	27
实例 10	DragOver 控件 .....	30
实例 11	TRadioButton 控件 .....	32
实例 12	TProgressBar 控件 .....	34
实例 13	Listbox 控件 .....	36
实例 14	Timer 控件 .....	39
实例 15	将数字转换为中文大写 .....	42
实例 16	在 Memo 控件中显示文件路径 .....	45
实例 17	设置系统热键 .....	47
实例 18	截取 Listbox 控件信息 .....	50
实例 19	TreeView 控件 .....	53
实例 20	TRichEdit 控件 .....	56
实例 21	实现超链接 .....	59
实例 22	Hook 程序记录和回放鼠标、键盘 .....	62

## 第二篇 Delphi 7 窗体设计

实例 23	透明窗体 .....	67
实例 24	渐变背景 .....	69
实例 25	控制 Windows 任务栏 .....	71
实例 26	制作字幕滚动窗体 .....	74

实例 27	Splash Screen.....	77
实例 28	查看所有当前系统窗口.....	81
实例 29	Windows 底层任务控制.....	85
实例 30	实时监控窗体尺寸.....	89
实例 31	伸缩窗体.....	93
实例 32	不规范窗体.....	96
实例 33	选择建立子窗体.....	99
实例 34	添加启动退出画面.....	102
实例 35	屏保预览.....	107
实例 36	特殊位置的按钮.....	111
实例 37	图片浏览工具.....	114
实例 38	特效窗体.....	122
实例 39	Office XP 风格的简单菜单设计.....	125

### 第三篇 Delphi 7 文件与系统

实例 40	文件的打开和关闭.....	129
实例 41	统计文本文件的中英文个数.....	132
实例 42	INI 文件读写的应用.....	135
实例 43	二进制文件的阅读.....	138
实例 44	文件查询.....	143
实例 45	文件的锁定保护.....	147
实例 46	骗人小程序.....	150
实例 47	多文档界面程序.....	152
实例 48	文件的关联操作.....	156
实例 49	文件夹信息查询.....	159
实例 50	文件创建及修改日期的查询.....	163
实例 51	应用程序快捷方式的制作.....	167
实例 52	建立软件注册机制.....	170
实例 53	测试 CPU 速度.....	174
实例 54	声卡检测.....	177
实例 55	查询内存信息状态.....	179
实例 56	磁盘驱动器信息.....	182
实例 57	Hook 程序中 DLL 文件的编写.....	185
实例 58	Hook 程序的测试.....	189
实例 59	映射网络驱动器.....	191

实例 60	状态栏中程序图标的显示 .....	194
-------	-------------------	-----

## 第四篇 Delphi 7 图形图像与多媒体

实例 61	获取图像的原色值 .....	200
实例 62	改变图像原色对比 .....	203
实例 63	Windows 标准动画 .....	207
实例 64	旋转图片 .....	211
实例 65	图像显示技巧（一） .....	215
实例 66	图像显示技巧（二） .....	219
实例 67	绘制基本图形 .....	223
实例 68	图形格式压缩转换 .....	226
实例 69	模拟放大镜效果 .....	230
实例 70	动态文字效果 .....	235
实例 71	图像鼠标过渡 .....	239
实例 72	随机图形 .....	242
实例 73	CaptureScreen 程序 .....	246
实例 74	图像镜像效果 .....	252
实例 75	运动小球 .....	257
实例 76	特殊图像效果 .....	261
实例 77	媒体播放器 .....	265

## 第五篇 Delphi 7 数据库编程

实例 78	ODBC 数据源的添加 .....	270
实例 79	SQL 语言自动生成器 .....	274
实例 80	数据表修改程序 .....	283
实例 81	数据库窗体向导的应用 .....	287
实例 82	数据库中导入文本文件资料 .....	290
实例 83	ADO 控件连接数据库 .....	294
实例 84	ADO 方法链接多个数据库 .....	298
实例 85	建立图像数据库 .....	302
实例 86	数据表中图像的显示 .....	306
实例 87	用户密码验证系统 .....	309
实例 88	数据库和数据表的自动建立 .....	313



## 第六篇 Delphi 7 网络技术

实例 89	获取 Mac 地址 .....	318
实例 90	信使服务 .....	321
实例 91	访问指定计算机的共享资源 .....	325
实例 92	ping 命令查验联网状态 .....	328
实例 93	局域网内共享文件资源的下载 .....	333
实例 94	实时监控邮箱新邮件 .....	335
实例 95	通过 SMTP 服务器发送邮件 .....	338
实例 96	IP 地址框的编辑 .....	341
实例 97	编写网页浏览器 .....	344
实例 98	创建聊天室 .....	348
实例 99	根据 IP 地址查计算机名 .....	352
实例 100	IP 地址和计算机名相互转化的多线程处理 .....	355

## 第一篇

# Delphi 7 编程基础及控件应用

作为 Inprise 公司的新一代产品, Delphi 7 不仅继承了 Pascal 语言的优点, 同时还开发提供了一整套用于设计、编写、测试、调试和发布应用程序的工具软件。而它对于 Microsoft 公司的 .NET 技术的支持使得 Delphi 7 的程序员可以更加随心所欲地开发各种软件。

控件是真正面向对象意义上的对象, 同时, 品种繁多的控件 (VCL) 是 Delphi 在众多编程软件中得以名声显赫的重要原因。正因为如此, Delphi 的控件封装了一些数据集和数据访问的过程与函数, 从父类中继承了数据和行为。尽管每个控件有其特殊性, 所有控件都共享从它们的共同祖先 Tcomponent 处继承来的某些属性。Tcomponent 定义了控件用于 Delphi 环境所必需的最小属性集。

控件以一种标准的可重用的方式封装了应用元素。了解 Delphi 提供的控件种类是了解 Delphi 的重要一步。

同时所有控件都有其内置的属性、方法和事件, 正确地设置和使用控件的属性、方法和事件是学习 Delphi 的关键。这些控件必须非常详细地掌握, 掌握得越熟练, 后面的学习就越容易。

本篇包括以下 4 个方面的内容: Delphi 编程基础、Delphi 基本控件、设置 Delphi 控件属性、控件方法与事件。结合大量实例详细说明了 Delphi 7 编程中的基本应用及新增技术, 以及如何设置并使用 Delphi 的各种控件及其属性, 并形象地解释和演示了 Delphi 控件的方法和事件的应用。

*Let's GO!* 

## 实例

## 1

## 加载图形按钮



## 实例说明

在 Delphi 中不仅可以创建标准按钮 (Button)，还可以创建其他各种类型的按钮。例如，可以创建加载图形的 BitBtn 按钮，同时在触发按钮事件时播放 WAV 声音文件。



## 技术指南

□ 编写本程序的过程中应用到了 3 个按钮事件：

OnMouseDown、OnMouseUp 和 OnClick。

□ 播放 WAV 文件需要调用 PlaySound 函数。

注意：播放声音文件需要触发 OnMouseUp 事件，而如果在按钮外释放鼠标将不能够触发 OnMouseUp 事件。





## 创作步骤

① 在 Delphi 7 中执行 File→New→Application 命令开启一个新的标准工程。

② 调整窗体大小，将 Caption 属性设为“一起来躲雨”。

③ 在 Additional 控件板中选择 BitBtn，为窗体添加一个 BitBtn 按钮，并将其 Caption 属性设置为“一起躲雨吧？”。

④ 单击选择 BitBtn 按钮的 Glyph 属性的  按钮，弹出 Picture Editor 对话框，在其中单击  并选择一个人躲雨的图片，如图 1-1 所示。

⑤ 添加完控件后的窗体如图 1-2 所示。

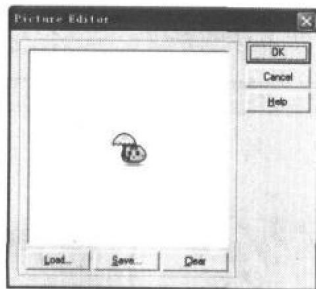


图 1-1 加载图像

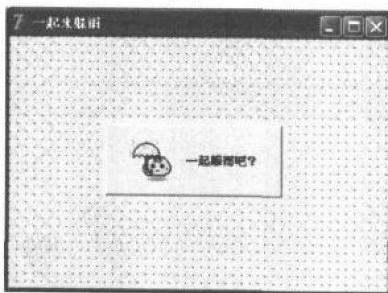


图 1-2 添加完控件后的窗体效果



## 程序代码

① 源程序代码如下:

```

unit FireForm;
interface
uses Windows, Classes, Graphics, Forms,
    Controls, Buttons, Dialogs, StdCtrls;
type
  TForm1 = class(TForm)
    BitBtnFire: TBitBtn;
    procedure BitBtnFireMouseDown(Sender: TObject; Button: TMouseButton;
      Shift: TShiftState; X, Y: Integer);
    procedure BitBtnFireMouseUp(Sender: TObject; Button: TMouseButton;
      Shift: TShiftState; X, Y: Integer);
    procedure BitBtnFireClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
{$R *.DFM}
uses
  MmSystem;
procedure TForm1.BitBtnFireMouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
  // 加载一个人躲雨
  if Button = mbLeft then
    BitBtnFire.Glyph.LoadFromFile ('duoyu2.bmp');
end;
procedure TForm1.BitBtnFireMouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
  // 加载两个人躲雨
  if Button = mbLeft then
    BitBtnFire.Glyph.LoadFromFile ('duoyu.bmp');
end;
procedure TForm1.BitBtnFireClick(Sender: TObject);
begin

```

```
PlaySound ('dingdong.wav', 0, snd_Async);  
MessageDlg ('大家要互相帮助哦!', mtWarning, [mbOK], 0);  
end;  
end.
```

② 具体控件属性设置见本书附带光盘。



## 结果演示

本例名为“一起来躲雨吧”。窗体中有一个简单的 BitBtn 按钮，按钮上是一个独自打伞的小人的图像，如图 1-3 所示。单击该按钮将弹出一个对话框，并在对话框中显示“大家要互相帮助哦！”的字样，同时播放一段 WAV 格式的声音文件，如图 1-4 所示。

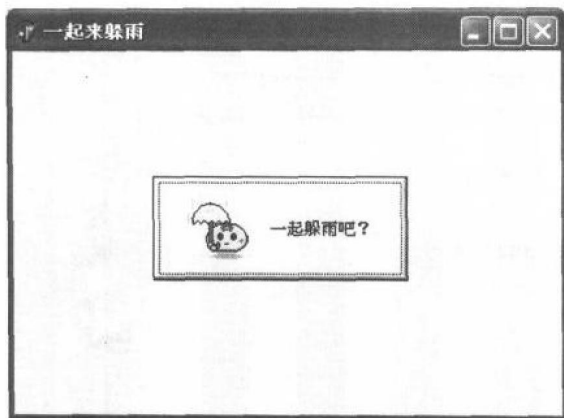


图 1-3 程序运行窗体

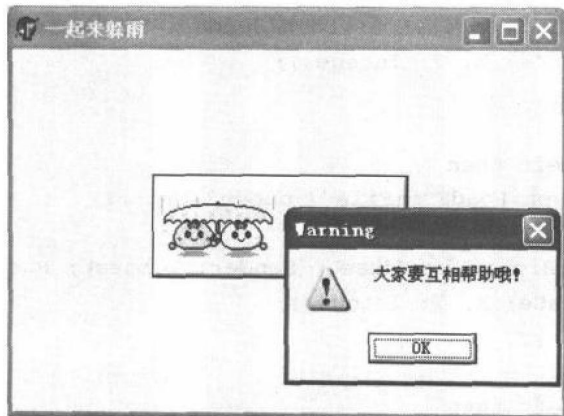


图 1-4 结果演示

## 实例

## 2

## 简易计算器



## 实例说明

利用 Delphi 的控件可以开发出各种类型的面板工具,这是 Delphi 一个很重要的功能组成。本例将制作一个简易计算器,实现基本的加减乘除运算功能。



## 技术指南

□ 本程序利用 Delphi 的 Button 控件、Label 控件和 Edit 控件实现面板的组成。

窗体上接收和显示数据都是利用 Edit 控件的 Text 属性,但它仅支持数据类型为 String 的变量,而四则运算要求数据是 Float 类型的,所以需要调用 StrtoFloat()函数和 FloatToStr()函数进行数据类型转换。

□ 为了使程序结构紧凑,本例使用 Case 语句实现加减乘除的运算。

□ 本例的关键代码在于执行计算,这就需要为“=”按钮的 OnClick 事件添加代码。其中,当进行除法运算时,如果除数为 0,需要给出错误提醒。运算完毕后,还需要将运算结果从运算框中显示出来。



## 创作步骤

① 在 Delphi 7 中执行 File→New→Application 命令开启一个新的标准工程。

② 激活 Form1 窗口,在 Object Inspector 中设置该窗体的 Caption 属性为“简易计算器”,Color 属性设置为 clBtnFace。

③ 在 Standard 控件板中选择 Edit 选项,在窗体生成一个 Edit 控件,用鼠标调整其合适大小和在窗体上的位置。再选择 Standard 页上的 Label 控件,将其放置在 Edit 控件左边,设置其 Caption 属性为“运算框”。

④ 在 Standard 控件板中选择 Button 选项,在窗体中添加 16 个按钮。分别选中各个按钮,在 Object Inspector 中将其 Caption 属性分别设置为“0”,“1”,“2”,“3”,“4”,“5”,“6”,“7”,“8”,“9”,“=”,“+”,“-”,“x”,“/”和“清空”。

⑤ 全选面板上所有的 Button 控件,并执行 Edit→Align 命令,弹出如图 2-1 所示 Alignment 对话框,从中选择合适的对齐方式,使控件在窗体上合适的布置,如图 2-2 所示。

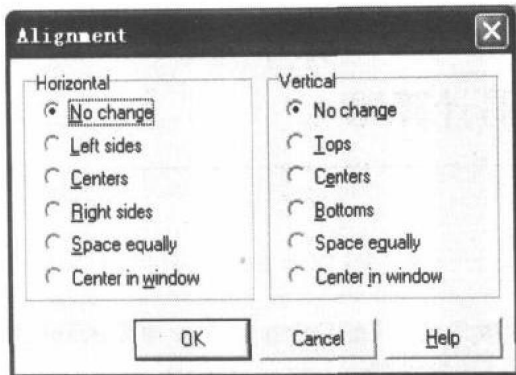


图 2-1 Alignment 对话框

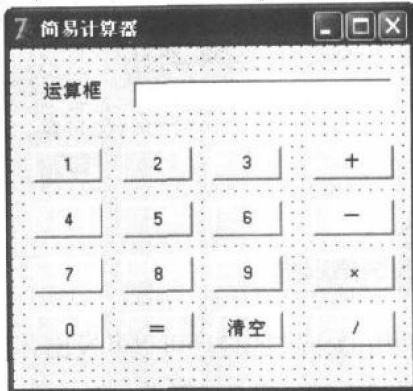


图 2-2 控件对齐后的窗体



## 程序代码

① 要实现按钮输入数据的功能，需要为每个按钮编写响应代码。为了实现这一功能，需要逐一为 0~9 的 10 个 Button 控件编写代码：

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    edit1.Text:=edit1.Text+'1'    //在运算框中输入数字"1"
end;
```

② 下面编写运算符控件的响应代码。当单击运算符时，给标志变量 n 赋值，然后将 Edit 控件中的数值赋值给 temp1 并将 Edit 清空，下面以加法为例介绍：

```
procedure TForm1.Button13Click(Sender: TObject);
begin
    n:=1;                //为加法运算设置标志变量
    temp1:=edit1.text;   //为临时变量 temp1 赋值
    edit1.clear;        //清空运算框
end;
```

③ 为“清空”按钮的单击事件添加代码很简单，只需用到 edit1.Clear 即可。

```
procedure TForm1.Button12Click(Sender: TObject);
begin
    edit1.Clear    //清除文本框中的数字
end;
```

④ 编写执行计算的代码时，用到了 Case 语句和 if 语句：

```
procedure TForm1.Button11Click(Sender: TObject);
begin
    case n of
        1:temp2:=strtofloat(temp1)+strtofloat(edit1.text);
        2:temp2:=strtofloat(temp1)-strtofloat(edit1.text);
        3:temp2:=strtofloat(edit1.text)*strtofloat(temp1);
```

```
4:if strtofloat(edit1.text)=0 then
begin
showmessage('除数为 0! ');
exit;
end
else
temp2:=strtofloat(temp1)/strtofloat(edit1.text);
end;
edit1.text:=floattostr(temp2);
end;
```

⑤ 完整代码和具体属性设置参见本书所附光盘。



### 结果演示

执行本程序将出现一个计算器窗口，并可在其中执行基本运算，如图 2-3 所示。

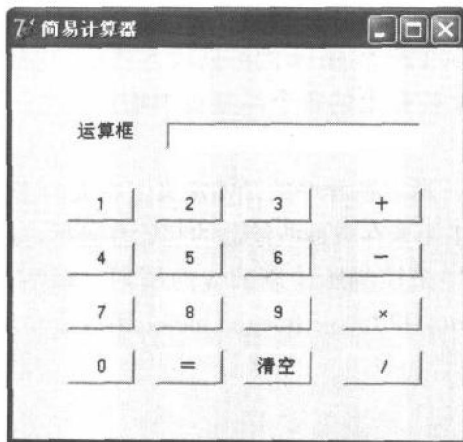


图 2-3 结果演示



## 实例

## 3

## 绘制多种曲线



## 实例说明

本例利用 Panel 和 Image 控件，通过对 Canvas 对象和 While 语句等的使用，创建了一个可以绘制多种数学曲线的程序。

同时，本例还利用 Delphi 7 中的 TabbedNotebook 控件建立多页标签，在其上根据需要显示的内容将窗体分成多个部分。




## 技术指南

- ☐ 利用 TabbedNotebook 的 Page 属性可以很方便地为窗体添加或删除标签页。同时，在 TabbedNotebook 控件上的每个标签页中均放置了一个 Image 控件，用以显示曲线。
- ☐ 由于绘制数学曲线，所以在程序的开始定义了相关的变量，它们是  $x_1$ 、 $x_2$ 、 $y_1$ 、 $y_2$  和  $n$ ，其中  $x_1$  和  $y_1$  用来存取曲线绘制的起始点坐标， $x_2$  和  $y_2$  用来存储曲线绘制的当前点，而  $n$  则用来控制所绘制曲线的周期。最后曲线的显示是通过调用语句 `Image.Canvas.MoveTo()` 和 `Image.Image.LineTo()` 函数来实现的。



## 创作步骤

- ① 在 Delphi 7 中执行 `File→New→Application` 命令开启一个新的标准工程。
- ② 激活 Form1 窗口，在 Object Inspector 中设置该窗体的 Caption 属性为“绘制曲线”。
- ③ 在窗体中添加一个 TabbedNotebook 控件（Win 3.1 控件板），单击 Object Inspector 面板中 Pages 属性右侧的  按钮，将弹出如图 3-1 所示的 Notebook editor 标签页编辑对话框。单击 Add 按钮可以添加标签页，单击 Edit 按钮可以编辑已有的标签页。选中某标签页后，单击 Delete 按钮可以删除该标签页。在这里一共添加了 4 个标签页，各个标签页的名称分别为“正弦曲线”、“余弦曲线”、“正切曲线”、“余切曲线”。
- ④ 通过单击 TabbedNotebook 控件的 Object Inspector 面板中的 ActivePage 下拉菜单中选取相应的标签页进行切换，分别对每个标签页添加一个 Panel 控件并在每个 Panel 控件中添加一个 Image 控件，所有 Image 控件的 Align 属性均设置为 alClient。
- ⑤ 在 TabbedNotebook 控件的每个标签页中添加一个 BitBtn 按钮，Caption 属性均设置为

