



C++ Builder

程序员学习数据结构

严仕伟
李 强

著
改编

内附光盘



人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

C++ Builder 程序员 学习数据结构

严仕伟 著

李 强 改编

人民邮电出版社

图书在版编目(CIP)数据

C++ Builder程序员学习数据结构/严仕伟著;李强改编.

——北京:人民邮电出版社,2003.7

ISBN 7-115-11215-0

I.C... II.①严... ②李... III.①C语言—程序设计 ②数据结构 IV.①TP312
②TP311.12

中国版本图书馆CIP数据核字(2003)第025011号

版 权 声 明

本书繁体字版名为《程序设计人员专业养成——使用C++ Builder》，由文魁信息股份有限公司出版，版权属严仕伟所有。本书简体字中文版由文魁信息股份有限公司委由KINGS INFORMATION CO.,LTD代为授权人民邮电出版社独家出版。未经本书原版出版者和本书出版者书面许可，任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部。

C++ Builder 程序员学习数据结构

- ◆ 著 严仕伟
改 编 李 强
责任编辑 陈冀康
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132705
北京汉魂图文设计有限公司制作
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 800×1000 1/16
印张: 19.5
字数: 438千字 2003年7月第1版
印数: 1-4000册 2003年7月北京第1次印刷

著作权合同登记 图字: 01-2003-0666号

ISBN 7-115-11215-0/TP·3414

定价: 36.00元(附光盘)

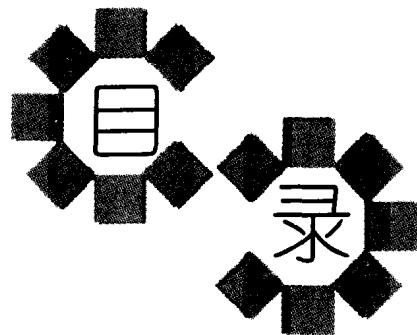
本书如有印装质量问题,请与本社联系 电话:(010)67129223

内 容 提 要

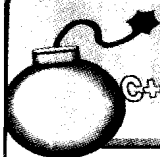
本书通过编程实践的方式帮助C++ Builder程序员学习和理解数据结构，以迅速提高程序设计能力。

全书共8章，讲解了递归方法、数组、堆栈、队列、链表、树、排序、搜索和图等基本数据结构的概念，并通过范例形象地演示了典型的应用。本书中的30多个完整的可执行范例，不仅能够帮助读者学习和理解，而且具有一定的实用价值；这些范例全部都收录在随书光盘中。附录部分给出了这些范例的操作说明，并且还给出了VCL的控件说明。

本书适合想要通过编程实例学习数据结构的C++程序员，也可以作为高校师生学习数据结构的参考书。



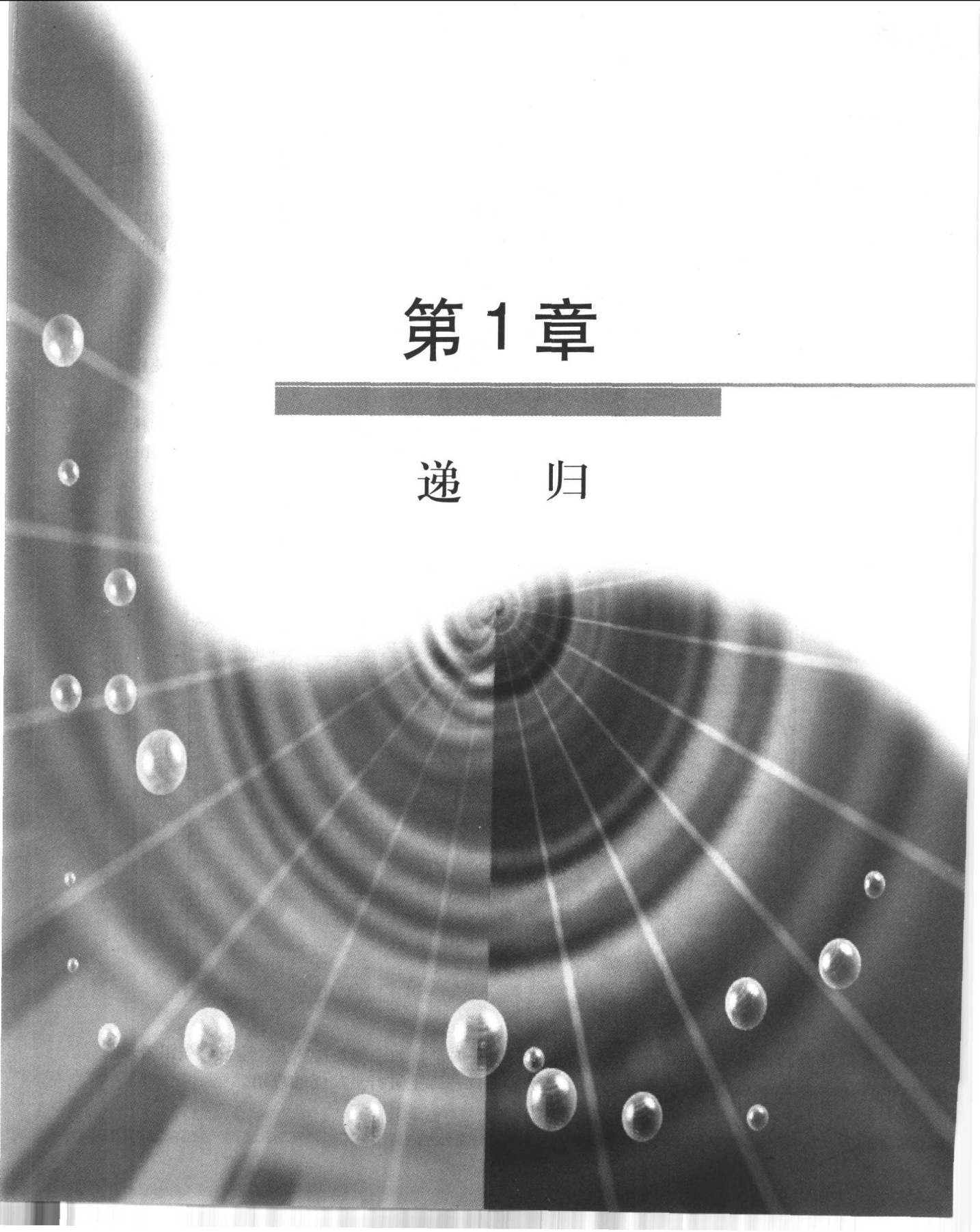
第1章 递归	1
范例1-1 简易计算器	3
第2章 数组	13
范例2-1 数组索引	14
范例2-2 工资存取	16
范例2-3 数组相乘	20
范例2-4 课表	24
第3章 堆栈与队列	30
范例3-1 基本堆栈	31
范例3-2 汉诺塔	37
范例3-3 基本队列	44
范例3-4 循环队列	52
第4章 链表	63
范例4-1 链表概念	64
范例4-2 链表模型	67
范例4-3 链表操作	75
范例4-4 双向链表	85
第5章 树	107
范例5-1 数组二叉树	110
范例5-2 结构二叉树	113
范例5-3 类别二叉树	120
范例5-4 二叉树应用	127
第6章 排序	132
范例6-1 起泡排序法	133
范例6-2 插入排序法	147
范例6-3 快速排序法	156
范例6-4 合并排序法	162
第7章 搜索	175
范例7-1 线性搜索法	176
范例7-2 二分搜索法	180

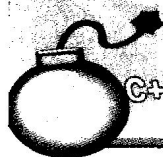


范例7-3 哈希搜索法	185
范例7-4 九宫棋	190
第8章 图	217
范例8-1 图表示	218
范例8-2 图搜索	224
范例8-3 最短路径	230
范例8-4 图的应用	238
附录A 范例程序操作练习	247
附录B VCL控件索引表	298

第 1 章

递 归





“数据结构”，顾名思义就是数据的结构，就好像是身体之中的骨骼，房屋之中的钢筋一样，其目的是构建程序内部数据的存储方法和相互关系。

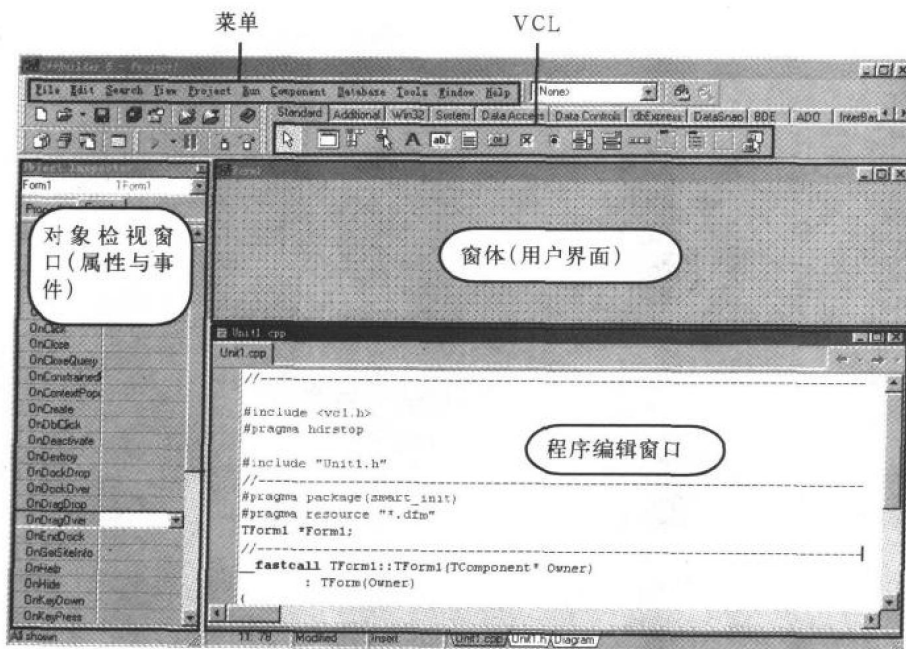
算法(Algorithm)是描述工作如何完成的步骤。简单地说，就是一个具有输入(input)输出、(output)且经过明确定义(Definiteness)的有限(Finiteness)且有效(Effectiveness)的步骤。

描述算法常用的方式有条列式、流程图、伪码及程序语句4种。

一个“好的程序”除了要有好的数据结构和算法外，注释可以帮助阅读及理解程序；定义有意义的变量名称能更清楚地表示程序算法处理的过程；程序缩排及段落的习惯，能使程序清楚并减少错误。简单地说，好的程序就像好的文章一样。

Borland C++ Builder (BCB)的设计理念，就像集成电路的设计可以用现有的IC模块快速地组建成电路一样。BCB已经内置了许多可视化控件库 (VCL, Visual Component Library)的集成开发环境 (IDE, Integrated Component Development Environment)。

单击“开始\程序\Borland C++ Builder 6\C++ Builder 6”或点选其快捷图标，就可打开如下图所示的IDE。



再多的说明都不如动手编写一个程序的印象深刻，如果您是初学者，建议您先跟着步骤做；如果您对此已比较熟悉，可以在范例程序的基础上进行更广泛的应用。



范例1-1 简易计算器

范例1-1是一个简单的计算器程序¹，在这个程序里我们将会认识到数据结构中递归(Recursive)的应用技巧(递归指一个过程或函数在其定义或说明中又直接或间接调用自身的一种方法，通常可以精简程序结构，从而增强程序的可读性与正确性)与BCB程序设计的轮廓，以下是程序运行结果：



执行步骤

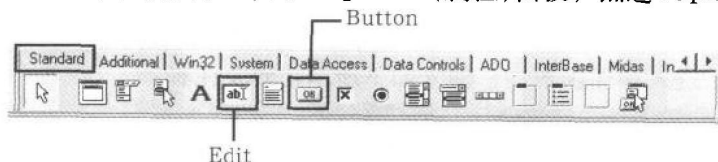
Step1 打开BCB的集成开发窗口环境(IDE)。

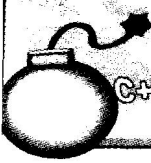
Step2 单击Object Inspector(对象检视)窗口中的Properties(属性)面板(该窗口中各项目是以英文字母顺序排列)。1.单击Caption(标题)并更改为“简易计算器”(注意窗体的变化)；2.接着单击Color(颜色)可根据下拉菜单点选颜色或双击鼠标左键自定义颜色；3.单击BorderStyle选择bsToolWindow。

Step3 1.在窗体上摆放VCL控件：单击控制选项Standard，再选取其中的VCL控件，并在窗体中适当位置点一下，就出现了一个你所选择的VCL控件；再在窗体上依次加入各个控件。

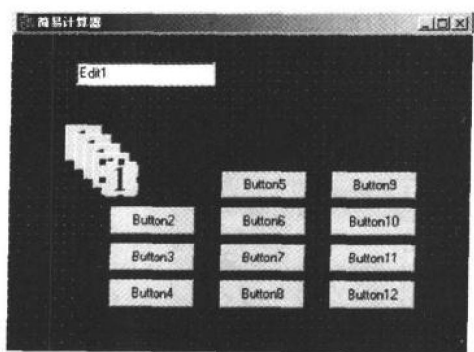
要选取多个控件时可按Shift键并同时点选控制项中多个控件，当控件被  框住时，即可连续在窗体中置入控件，如果有很多相同的对象类型或属性，可以先做好一个后再复制。

2.选取窗体中原始对象Button1(被选取时对象上会出现可调整大小的黑色方框)，再点选Object Inspector(对象检视)窗口中的Properties(属性)面板，点选Caption(标题)并更改





为“1”，接着单击Font(字体)，将字体大小设为24。3.确定选取到对象，按Ctrl+C键复制，再按Ctrl+V粘贴，各窗体成员的状态如下图所示：



Step4 规划窗体及控件的属性设置值，可参见下表：

Step5 依次更改各控件的属性设置值：

名 称	属 性	设 置 值 输 入
Form1	Name (只能用英文)	默认为Form1
	Caption (可用中文)	简易计算器
Edit	Name	eout
	Text	空白
Button1 ~ Button10	Name	依次更改为c0~c9
	Caption	依次更改为0~9
	Font	字体大小设为24(可按自己喜好试试看)
其他Button	Name	简单易记的名称为佳
	Caption	参考程序结果

先选取窗体或对象，Object Inspector窗口就会依照对象而跟着变化，找到属性后单击它，再按照规划值更改即可。一定要熟悉控件(VCL)与Object Inspector窗口搭配使用。

Step6 写入事件程序代码：

双击“0”按钮，或单击它并在Object Inspector内单击Events事件列表，再选取OnClick事件并双击鼠标，进入事件函数内，输入如下所示代码及注释：



注

双斜线(//)后以及(/*)、(*)/之间都是注释,不会影响程序的运行。

如输入错误可点选菜单上的Edit/Undo命令恢复。

```
//-----
void __fastcall TForm1::c0Click(TObject *Sender)
    //注意,是c0的OnClick事件
{
    String c0="0";    /*设置c0这个Button控件的数据类型为字符串类型
                       (string),内容为0*/
    c=c+c0;           /*累计数字,合并字符串,(=)是赋值的意思,就是将
                       c字符串与c0字符串合并后再返回给c字符串*/
    eout->Text=c;      /*将c字符串内容赋给名称为eOut(Edit对象)的
                       Text属性内容,->是由减号(-)与大于号(>)组成*/
    f=StrToFloat(c);  /*将c由字符串类型(Str)转换成(To)浮点类型
                       (Float),再赋值给f*/
}
//-----
```

将程序编辑窗口的滚动条拉到最上方,并输入如下代码及注释:

```
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;        //将变量设于此处就可以用于所有窗体事件中
String c;              /*将变量c设为字符串类型(String),注意在C语言中
                       字母大小写是不同的*/
float f,f1,f2;         /*将f,f1,f2设为浮点类型(float),常用变量类型
                       请参考附录*/
int opr,vrb;           //将opr这个变量设为整数类型(int)
                       /*operators(opr),variable(vrb)变量名称除
                       简易原则外,最好是具有意义的词*/
//-----
```

点选窗体,分别双击1~9按钮;或单击它们并在Object Inspector内单击Events事件列表,再选取OnClick事件并双击鼠标,进入事件函数内,编辑程序:

注

0~9的按键程序内容除变量外其他都一样,可按鼠标右键使用Copy与Paste指令。

复制编辑好的c0事件程序,贴在1~9 OnClick事件中(重复上述点入粘贴操作9次)。

```
//-----
```

```

void __fastcall TForm1::c8Click(TObject *Sender)
    //请不要随意复制或更改函数名称
{
    //请双击对象事件后进入函数
    String c8="8";
    //应复制的是大括号中的内容
    c=c+c8;
    eout->Text=c;
    f=StrToFloat(c);
}
//-----
void __fastcall TForm1::c9Click(TObject *Sender)
{
    String c9="9";
    /*粘贴函数内容后,再根据控件事件的不同而
    修改变量*/
    c=c+c9;
    eout->Text=c;
    f=StrToFloat(c);
}
//-----

```

接下来设置加减乘除功能的条件值,双击“+”按钮,进入程序编辑窗口,输入下列程序:

```

//-----
void __fastcall TForm1::c01Click(TObject *Sender)
    //"+"对象名称设为c01
{
    c=" ";
    //将c字符串设置为空
    opr=1;
    //将变量opr的值设置为1
    f1=f;
    //f1为取得的第1个数值,其值是从c字符串所转换的数值f
    f=0;
    //将f设为0以便取得第2个数值
}
//-----

```

“-”、“*”、“/”按钮的程序内容除变量opr设置值外其他都一样,可按鼠标右键使用Copy与Paste指令复制编辑好的c01事件程序,贴在2~4 OnClick事件中(重复上述点入粘贴操作3次)。

设置正负号转变的功能,双击“+/-”按钮,进入程序编辑窗口,输入下列程序:

```

//-----
void __fastcall TForm1::Button13Click(TObject *Sender)
{
    f=-f;
    //改变f变量的正负值
    eout->Text=FloatToStr(f);
    //将变量f由浮点类型转换成字符串,
    并设置给eout输出
}
//-----

```



接下来设置清除按钮双击“C”按钮，进入程序编辑窗口，输入下列程序：

```
//-----
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    c=" ";
    eout->Text=" ";
    f=0, f1=0, f2=0;
}
//-----
```

设置结果演算及输出程序，双击“=”按钮，进入程序编辑窗口，输入下列程序：

```
//-----结果演算及输出设置-----
void __fastcall TForm1::resClick(TObject *Sender)
{
    int fs(int,int);           //求次方的函数定义
    int gcd(int,int);         //求最大公因数的函数定义
    int t;                     //将变量t定义为整数型
    f2=StrToFloat(eout->Text); //截取第2次输入显示的值
    eout->Text=" ";            //将字符串内容清空以便于结果输出
    c=" ";
    switch(opr)                //switch选择语句,以opr变量内容为选择条件
    {
        case 1:                //当opr内容值为1时运行下列语句
            f=f1+f2;            //将第1次输入的值加第2次输入的值赋给f变量
            break;              //跳出switch选择语句
        case 2:
            f=f1-f2;
            break;
        case 3:
            f=f1*f2;
            break;
        case 4:
            f=f1/f2;
            break;
        case 5:                //乘方的opr内容值为5
            f=fs(f1,f2);        //乘方的递归函数结果值,赋给f变量
            break;
        case 6:                //最大公因数的opr内容值为6
            if(f2>f1)           //如果(if) f2的值大于f1则运行大括号中语句
            {
                t=f1; f1=f2; f2=t; //将f1与f2的值互换
            }
            f=gcd(f1,f2);       //最大公因数的递归函数结果值,赋给f变量
    }
}
```

```

        break;
    }
    eout->Text=FloatToStr(f);    /* 将switch选择运行结果转成字
                                   符串,设置给eout输出*/
}
//-----

```

接下来,有4个以递归方式编写的程序,第1个为方根、第2个为求乘方、第3个为求阶乘、第4个为求最大公因数;在第1个递归范例中,我们以二分猜值算法求平方根的解,在开始编写程序之前,我们应该先设计递归函数的内容:

1. 函数的输入值与输出值之间的关系(例:求平方根的解,输入9则输出为3);
2. 设置结束终止条件;
3. 确认过程与结果的正确性。

以下是第1个递归程序范例求平方根的程序代码:

```

//-----以二分猜值算法求平方根的解-----
int sr(int x,int y,int z)
    //接收sr(s,b,b)的值,最后返回一个整数值
{
    int m;           //定义中间值m为整型
    if(abs(x-y)<=1)
    //如果(if),x-y的绝对值(abs)小于等于(<=)1时,<<这是结束条件>>
        return (x);    //运行返回x值的动作 <<注意x是最后答案>>
    m=(x+y)/2;        //将x+y的值,除以2后,赋给m
    if(m*m-z <=0)
    //如果(if) m的平方小于z(就是b的传递值),则猜的值过小
        sr(m,y,z);    //运行将m值设为最小边界值并返回事件函数
    else
        //否则运行下列语句
        sr(x,m,z);    //猜的值过大则将m值设为最大边界值并返回事件函数
}
//---程序运行是由事件函数开始的,递归函数若写在上方,就可以不做定义直接使用---
void __fastcall TForm1::Button1Click(TObject *Sender)
    //平方根按钮OnClick事件函数
{
    //也可直接引用C++内置函数库math数值函数f=sqrt(f)求解
    int s=0,b;        //定义两个变量s为最小边界值,直接设定为0;b为最大边界值
    b=f;              //将f设置给最大边界值b
    f=sr(s,b,b);      //sr()为递归函数,将结果值传给f变量
    eout->Text=FloatToStr(f); //输出演算结果
}
//-----

```

求平方根的程序运行范例如下:



點選輸入值：本范例為9。

輸出結果即為3：



第2个范例是求乘方的递归程序，求乘方要输入两个变量，其取值方法与加法相同，双击代表次方的按钮，进入事件程序编辑窗口，并输入下列程序代码(可剪贴加法事件程序并加以修改)：

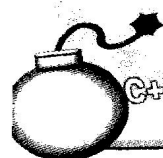
```
//-----
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    c=" ";
    opr=5;           //乘方解的opr变量设置为5
    f1=f;
    f=0;
}
//-----
```

其递归函数与起始调用将在等号按钮的OnClick事件中设置，而主要的递归程序代码则在等号按钮的OnClick事件函数后编写，程序代码如下：

```
//-----求乘方解的递归函数-----
int fs (int m,int n)

//等号事件函数,switch选择中,csae 5:相互调用的递归函数
//注意使用前需先定义,请看等号的事件程序开头
//当次方n为0的时候返回1<<结束条件>>
//程序语句以分号(;)为段落,当语句超过两行时就需加大括号{}
//当次方数为1的时候返回m值
//否则返回m*fs(m,n-1),此处为递归演算法的重点
//结果为m相乘n次

{
    if(n==0)
        return 1;
    if(n==1)
        return(m);
    else
        return(m*fs(m,n-1));
}
//-----
```



第1个输入值：9(基数)，然后点选计算次方的功能按钮。



第2个输入值：3(次方数)，计算器输出值：729(即 $9*9*9$)。



第3个介绍的递归程序是求阶乘，求阶乘的程序码如下：

```
//-----递归方式求阶乘-----  
int fa(int n)           //接收值为整数型的n变量,返回值为整型  
{  
    if(n==0)            //如果n等于0,请注意等于为(==)  
        return 1;      //递归结束条件  
    else  
        return(n*fa(n-1)); //主要递归算法  
}  
void __fastcall TForm1::c08Click(TObject *Sender)  
    //阶乘按钮OnClick事件函数开始  
{  
    f=fa(f);            //将递归函数结果值赋给f变量
```



```
eout->Text=FloatToStr(f);    //显示f值
}
//-----
```

求阶乘时仅输入一个值9，然后单击求阶乘的功能按钮即可产生输出：



第4个递归函数是求最大公因数，此递归程序需要输入两个变量（其取值方法和加法相同）。双击求最大公因数的按钮，进入事件程序编辑窗口，并输入下列程序代码或剪贴加法事件中的程序代码：

```
//-----
void __fastcall TForm1::c06Click(TObject *Sender)
{
    c=" ";
    opr=6;                //求最大公因数的opr变量值为6
    f1=f;
    f=0;
}
//-----
```

其递归函数与起始调用将在等号按键OnClick事件中设置，而主要的递归程序代码则在等号OnClick事件函数后编写，程序代码如下：

```
//-----最大公因数的递归函数-----
int gcd(int a,int b)
{
    //等号事件函数,switch选择中,csae 6:相互调用的递归函数
    //注意使用前需先定义,请看等号的事件程序开头
    if(b==0)                //如果b值等于0,返回a值<结束条件>
        return a;
    else                    //否则返回gcd(b,a%b);
        return(gcd(b,a%b)); //用的是反复相除的算法
}
//-----
```