

时尚百例丛书

**100**  
时尚编程百例



# C++

# 时尚编程百例

 机械工业出版社  
China Machine Press

●网冠科技 编著

100

时尚百例丛书

# C++ 时尚编程百例

网冠科技 编著

光盘包含本书素材、效果文件



机械工业出版社

C++程序设计语言是在 C 语言的基础上发展起来的一种新的面向对象的程序设计 (Object-Oriental Programming) 语言, 是开发底层应用程序的重要工具之一。它包含了 C 语言的所有功能, 并扩充了许多新功能 (如函数重载、封装性、继承性、多态性等)。

本书的内容包括: C++语言不同于 C 语言的新功能和新特性, C++语言的面向对象方法和设计思想, 用 C++语言开发的一些图像和音乐程序, 用 C++语言编写的游戏, 以及用 C++语言处理操作系统中的一些常见问题和实用程序等。

本书附带包含书中 100 个实例的可执行程序 and 源代码的光盘一张, 既可作为不同层次培训班的教材, 也可作为供广大 C++语言爱好者自学 C++语言和程序设计的书籍, 具有很大的实用价值。

## 图书在版编目 (CIP) 数据

C++时尚编程百例 / 网冠科技编著.

-北京: 机械工业出版社, 2001.7

(时尚百例丛书)

ISBN7-111-09089-6

I. C … II. 网… III. C 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2001) 第 044296 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 王 虹

责任印制: 路 琳

北京市密云县印刷厂印刷 · 新华书店北京发行所发行

2001 年 7 月第 1 版 · 第 1 次印刷

787mm × 1092mm  $\frac{1}{16}$  · 21.5 印张 · 529 千字

0001-6000 册

定价: 38.00 元 (含 1CD)

凡购本图书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话: (010) 68993821、68326677-2527

## 出版说明

随着 21 世纪的到来,人们更深切地感受到了计算机在生活和工作中的作用越来越重要,越来越多的职业需要具有计算机的应用技能。掌握计算机是职业的需要,更是事业发展的需要。

目前计算机技术不但广泛地应用在办公自动化中,它还全面渗透到各行各业。如果要从从事平面设计的相关行业,就应该学会平面设计软件,如 Photoshop、CorelDRAW、FreeHand 等;如果要从从事三维设计的相关行业,就应该学会三维设计软件,如 3DS MAX、Maya、Poser 等;如果要从从事多媒体设计的相关行业,就应该学会多媒体制作软件,如 Authorware、Director、Premiere 等;如果要从从事与网络相关的行业,就应该学会 Flash、Dreamweaver、Fireworks、ASP、PHP、JavaScript 等;如果要从从事建筑产品、工业产品设计的相关行业,就应该学会 AutoCAD、3DS VIZ、Protel 等;如果要从从事软件开发的相关行业,就应该学会 VB、VC、VFP、Delphi、PowerBuilder 等编程。

所有与计算机相关的职业都要求工作者有很强的计算机操作技能,做到运用自如,熟练而且深入地掌握软件的应用。而要做到这一点,必须从软件的各个方面入手,通过实例演练的方式训练自己,而且要反复练习,做到举一反三。

为了让读者能深入而且熟练地掌握相关软件的应用方法,机械工业出版社特别为广大读者推出了这套时尚百例丛书。本丛书对每一个应用软件精心制作了 100 个实例,其宗旨就是让读者全方位掌握软件的应用,为广大读者提供一条快速掌握计算机应用技能的捷径。

本丛书采用新颖的版式,将知识和实例紧密结合,通过对各种实例的详细讲解,使读者不必事先学习各种软件,而从实例的制作过程中体会到每个软件每项功能的使用方法,并自己做出各种实例效果,这样既节省了大量时间,同时也使读者有身临其境的感觉,并可以反复演练,将所学知识运用到职业工作中去。

书山有路勤为径。愿广大读者能通过本丛书的学习掌握计算机技能,并应用到自己的工作和事业中去。

机械工业出版社



# 前 言

《C++时尚编程百例》是“时尚百例丛书”中的一本。

C++语言是一种混合性语言，早在1980年就由AT&T贝尔实验室开发，迄今仍在演变之中。C++语言的诞生具有重要的意义，它标志着人们的程序设计从传统的“过程化程序设计”转入新的“面向对象程序化设计”(Object-Oriental Programming)；同时，C++语言比支持面向对象的Ada和Modula-2更加前进了一大步，并保留了C程序设计语言的简洁性和高效性。

C++语言程序设计以灵活强大的函数库和OOP功能受到广大数据库开发人员的欢迎。OOP的主要特点有以下几个方面：第一，“封装性”；保证数据结构和相应的方法的安全性和保密性；第二，“继承性”；保证程序可以避免重复的编写工作，使程序具有良好的可移植性；第三，“多态性”；使不同层次体系的类(Class)可以有相同名称但不同功能的方法(Method)。OOP为开发人员提供了良好的机制，为他们编制功能强大、内容丰富的应用软件提供了便利的条件。另外，C++语言的函数库也为程序员编写应用程序节省了精力和时间。


本书以实例的形式向读者讲解C++语言的基础知识，并教会读者如何使用C++语言开发一些应用程序。全书制作了100个实例，分为：基础篇，图形篇，游戏篇和应用篇四个部分，以满足不同层次读者的需要。对于初学者，通过本书的基础篇学习，可以迅速了解C++语言的功能和特性，并可以快速创建自己的应用程序，从而可以熟悉掌握C++语言的使用；对于已经有一定基础的中级水平的开发人员，可以通过本书的实例学习到一些具有实际意义的开发方法，进一步提高自己的开发应用程序的能力；对于高级开发人员，本书讲解了许多处理问题的算法和技巧，适合高级人员参考。

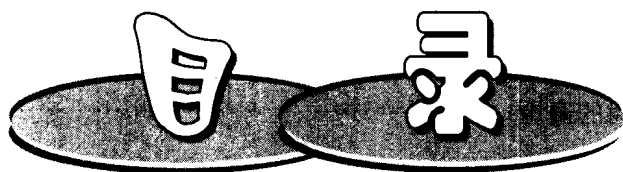
相信通过本书100个实例的学习，可以提高您开发C++语言应用程序的效率。真诚希望我们的讲解对您有所帮助。

本书由朱岩、刘晖编写完成，不当之处，还望专家读者批评指正。



网冠科技

本书光盘含配套素材（使用方法请见光盘中“光盘使用说明书”），技术支持请点击网冠科技站点 [Netking.163.com](http://Netking.163.com)。E-mail: [Netking\\_@yeah.net](mailto:Netking_@yeah.net)。  是网易公司的标志。



## 出版说明 前言

### 第一篇 基础篇

实例 1	中国你好	2
实例 2	全局变量和局部变量	4
实例 3	结构的使用	6
实例 4	指针和引用的使用	8
实例 5	类型转换	12
实例 6	三角函数	14
实例 7	取得随机数	16
实例 8	大小写的转换	18
实例 9	同名函数	21
实例 10	内嵌式函数	24
实例 11	传递参数的新方式一	26
实例 12	传递参数的新方式二	28
实例 13	新运算符与流	30
实例 14	字符串处理	32
实例 15	构造函数	35
实例 16	析构函数	39
实例 17	作用域限定运算符	42
实例 18	运算符重载	44
实例 19	操作符重载函数	48
实例 20	类的自引用	51
实例 21	类的导出	54
实例 22	多重继承的应用	60
实例 23	虚拟方法	63
实例 24	类的嵌入	66
实例 25	链表的定义与操作	69
实例 26	有序链表的生成和插入	76
实例 27	双向链表	81



实例 28	栈	87
实例 29	队列	92
实例 30	二叉树的建立与搜索	97
实例 31	二叉树的遍历	104

## 第二篇 图形篇

实例 32	直线的绘制	113
实例 33	模拟星系	118
实例 34	旋转的线	121
实例 35	自定义线的样式	123
实例 36	简单的小房子	126
实例 37	一个小雪人	129
实例 38	小小友好界面	132
实例 39	用户自定义填充模式	135
实例 40	蜜蜂窝	137
实例 41	模拟立体球	139
实例 42	光环的绘制	141
实例 43	圆形与扇形	144
实例 44	立体小精灵	146
实例 45	条形与立体条形	148
实例 46	仿真窗口	150
实例 47	模拟的时间条	153
实例 48	简单的小动画一	155
实例 49	简单的小动画二	157
实例 50	简单的友好界面	160
实例 51	字符的颜色	162
实例 52	图形方式下字符的设置	164
实例 53	立体窗口	166
实例 54	文本方式下的窗口	168
实例 55	图形方式下的窗口	171
实例 56	清屏方法一	173
实例 57	清屏方法二	177
实例 58	清屏方法三	179
实例 59	清屏方法四	181
实例 60	清屏方法五	183
实例 61	清屏方法六	185
实例 62	清屏方法七	187

实例 63	清屏方法八	189
实例 64	清屏方法九	191
实例 65	清屏方法十	193
实例 66	屏幕特效一	195
实例 67	屏幕特效二	199
实例 68	改变调色板	202
实例 69	霓虹闪烁	205
实例 70	蓝色的球	207
实例 71	设置 256 种颜色一	210
实例 72	设置 256 种颜色二	214
实例 73	一个亮点	216
实例 74	立体图形初步	218
实例 75	3D 制作初步	220

## 第三篇 游 戏 篇

实例 76	数据转条形图	224
实例 77	数据转线条图	226
实例 78	显示时间	228
实例 79	闹铃	230
实例 80	钟表	232
实例 81	生日快乐歌	234
实例 82	计时器	236
实例 83	费波那契数列	239
实例 84	三原色问题	242
实例 85	人机苹果大战	246

## 第四篇 应 用 篇

实例 86	获得当前路径	254
实例 87	修改当前工作路径	256
实例 88	菜单	258
实例 89	温度的转换	261
实例 90	比较排序法	263
实例 91	快速排序	265
实例 92	DOS 版本显示	268
实例 93	磁盘的组成	270



实例 94	显示模式切换 .....	272
实例 95	文件管理程序 .....	274
实例 96	院校管理系统 .....	281
实例 97	时间片轮转 .....	289
实例 98	进程的页式分配 .....	300
实例 99	进程的执行和换入换出一 .....	310
实例 100	进程的执行和换入换出二 .....	323

# 第一篇

## 基础篇

### 本篇总览

本篇主要介绍 C++ 语言中的一些基础知识，主要包括两个方面：一方面是 C++ 语言比传统的 C 语言所多出的功能和特性（非 OOP 方面），如函数的重载功能、new 及 delete 运算符等；另一方面主要介绍 C++ 语言所具备的面向对象功能（OOP 方面）及其相关的应用，如封装性，继承性，多态性等。

通过本篇中各个实例的制作，读者可以清晰地了解及熟练掌握 C++ 语言中的各种基本的功能和特性，并且对今后自己开发一些基本的应用程序有一定的指导作用，使读者能够更好地进行 C++ 语言程序的开发工作。

## 实例 1 中国你好

## 实例说明

本例的运行效果如图 1-1 所示。这是一个显示基本的文本字符串的应用程序。

本实例的知识点有：C++语言中显示字符串的方法和结构指针常量定义的方式。



图 1-1 效果图

## 编程思路

本实例的创作主要是为了帮助读者建立常量可以定义为任意类型的概念；显示方法。

关于程序实例本身的设计思路，主要定义了屏幕结构常量指针（SCREENPOINTER），而这个指针的类型为“SCREENARRAY”类型，定义了屏幕 25\*80 的字符显示；通过设定“\*SCREENPOINTER=(SCREENARRAY far \*)0xb8000001”，把“SCREEN”和屏幕显示直接联系起来。结构体“STRUCT”本身的两个域分别表示：显示的字符和字符的颜色，通过两个“FOR”循环，将“HELLO CHINA”的字符串值赋予 SCREEN[ROW][COLUMN]变量。

通过鼠标左键双击可以执行光盘中实例 1 的可执行文件“Screen.exe”，结合预先设定的显示结果进行观察。

## 创作步骤

程序的各个部分功能说明和主要源代码如下（用于开篇的第一个实例——结构指针常量）：

```
//THE PROGRAM IS TO TEST THE SCREEN POINTER'S USING  
//FILE SCREEN.CPP
```

## 1. 定义屏幕结构常量指针

```
#define SCREEN (*SCREENPOINTER)
```

## 2. 定义行列常量

```
#define ROW 25
```

```
#define COLUMN 80
```

### 3. 系统库函数调用

```
#include <string.h>
#include <stdio.h>
#include <conio.h>
```

### 4. 定义“TEXT”结构体

```
typedef struct TEXT
{
    unsigned char CHAR;
    unsigned char ATTRIBUTE;
}TEXT;
```

### 5. 定义整个字符显示屏幕的结构数组

```
typedef TEST SCREENARRAY[ROW][COLUMN];
```

### 6. 建立和屏幕的联系

```
SCREENARRAY far *SCREENPOINTER=(SCREENARRAY far *)0xb8000000l;
```

### 7. 定义要显示的常量字符串

```
char HELLO[]="HELLO CHINA!";
```

### 8. 主程序

```
int main(void)
{
    int CIRCLE; //定义循环变量
    clrscr(); //清屏
    for(CIRCLE=0;CIRCLE<strlen(HELLO);CIRCLE++) //第一次显示字符
    {
        SCREEN[10][CIRCLE].CHAR=HELLO[CIRCLE];
    }
    getch(); //观察结果（变色前）
    for(CIRCLE=0;CIRCLE<strlen(HELLO);CIRCLE++) //变换颜色属性
    {
        SCREEN[10][CIRCLE].ATTRIBUTE=11;
    }
    getch(); //再次观察结果并正常返回
    return 0;
}
```

## 实例 2 全局变量和局部变量

### 实例说明

本例的运行效果如图 2-1 所示。这是一个演示全局变量与局部变量区别的应用程序。

本实例的知识点有：C++语言中全局变量与局部变量的区别，以及使用方法。



图 2-1 效果图

### 编程思路

本实例的创作主要是为帮助读者学会如何设置全局变量和局部变量。

- (1) 全局变量一旦被定义，就可以在所有的函数中使用。
- (2) 局部变量定义之后，只能在这个变量所在的函数中使用。

由于程序本身比较简单，请读者自己结合实例的注释自行分析程序。

通过鼠标左键双击可以执行光盘中实例 2 的可执行文件“Variable.exe”，并且注意程序内的注释，按照注释中的提示自己对程序进行修改后，再进行编译，观察是否能够正常地通过编译程序。

### 创作步骤

程序的各个部分功能说明和主要源代码如下：

#### 1. 系统库函数调用

```
#include <conio.h>
#include <stdio.h>
```

#### 2. 定义全局变量

```
int GLOBAL;
```

#### 3. 子程序——测试程序

```
void TEST();
```

#### 4. 主程序

```
int main(void)
{
    clrscr();                //清屏
    TEST();                 //调用子程序，修改全程变量，输出局部变量的值
    printf("\nGLOBAL=%d",GLOBAL);
//注意：下面的语句不合法，因为 LOCAL 为局部变量，不能在 TEST()之外使用。
//自己动手：将此语句的注释去掉，重新编译，注意结果。
//printf("\nLOCAL=%d",LOCAL);
    getch();
    return 0;
}
```

### 5. 子程序——定义局部变量并修改全局变量

```
void TEST()
{
    int LOCAL;              //定义局部变量
    LOCAL=10;
    printf("\nLOCAL=%d",LOCAL);
    GLOBAL=20;             //修改全程变量的值
}
```

## 实例3 结构的使用

## 实例说明

本例的运行效果如图 3-1 所示。这是一个用结构 (STRUCT) 处理学生系统的应用程序。

本实例的知识点有: C++语言中结构的使用方法。

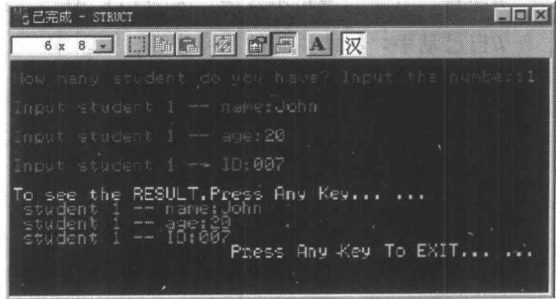


图 3-1 效果图

## 编程思路

本实例将介绍如何使用 C++语言中的结构 (STRUCT)。

(1) 结构的特点是: 可以表示一组不同类型的数据。而这组数据经常具有一定的相关性。

(2) 使用结构时, 表示结构中的成员的方法是: 结构变量名 . 结构成员。其中“.”是运算符。

通过鼠标左键双击可以执行光盘中实例 3 的可执行文件“Struct.exe”, 可以直接观察程序的运行结果。

## 创作步骤

程序的各个部分功能说明和主要源代码如下:

## 1. 系统库函数调用

```
#include <conio.h>
#include <stdio.h>
```

## 2. 主函数

```
void main()
{
    int NUM,I;
    struct STUDENT                //定义结构
    {
        char NAME[30];            //姓名
        int AGE;                  //年龄
        char ID[10];              //学号
```

```
};  
struct STUDENT STD[100]; //使用刚才定义的结构类型  
clrscr(); //清屏  
textcolor(LIGHTBLUE); //要求输入学生的个数 (NUM)  
cprintf("\nHow many student do you have? Input the number:");  
scanf("%d",&NUM);  
textcolor(CYAN); //按学生的个数输入学生的记录  
for(I=1;I<=NUM;I++)  
{  
cprintf("\nInput student %d -- name:",I); //输入学生的姓名  
scanf("%s",&STD[I-1].NAME);  
cprintf("\nInput student %d -- age:",I); //输入学生的年龄  
scanf("%d",&STD[I-1].AGE);  
cprintf("\nInput student %d -- ID:",I); //输入学生的学号  
scanf("%s",&STD[I-1].ID);  
}  
textcolor(YELLOW); //输出结果  
cprintf("\nTo see the RESULT,Press Any Key... ..");  
getch();  
textcolor(WHITE);  
for(I=1;I<=NUM;I++)  
{  
printf("\n student %d -- name:%s",I,STD[I-1].NAME); //输出学生的姓名  
printf("\n student %d -- age:%d",I,STD[I-1].AGE); //输出学生的年龄  
printf("\n student %d -- ID:%s",I,STD[I-1].ID); //输出学生的学号  
}  
textcolor(YELLOW);  
cprintf( "\nPress Any Key To EXIT... .."); //观察结果并返回  
getch();  
}
```



## 实例4 指针和引用的使用

## 实例说明

本例的运行效果如图 4-1 所示。这是一个演示普通型变量、指针以及引用型（Reference）变量的使用方法的程序。

本实例的知识点有：C++语言中变量的使用方法。

```

POINTER
自动
INPUT A NUMBER:1
INPUT A NUMBER:2
X= 1
Y= 2
X=1 Y=2
X=0 Y=0
X=3 Y=4
X=0 Y=0
  
```

图 4-1 效果图

## 编程思路

本实例将介绍如何使用 C++语言中的变量（VARIABLE）。

C++语言中的变量大致可以分为 3 种：普通型变量，指针型变量，引用型变量。

(1) 普通型变量 基本的数据类型：

int 整数  
char 字符  
float 单精度浮点数  
double 双精度浮点数  
enum 枚举

不同的数据类型，在不同的说明方式下，长度（所占的位数）和表示数据的范围也都有所不同，详见表 4-1。

表 4-1 基本数据类型

类型	所占位数	表示数据范围
unsigned char	8	0~255
char	8	-128~127
enum	16	-32768~32767
unsigned int	16	0~65535
short int	16	-32768~32767
Int	16	-32768~32767
unsigned long	32	0~4294967295
Long	32	-2147483648~2147483647
Float	32	3.4E-38~3.4E38
double	64	1.7E-308~1.7E308
long double	80	3.4E-4932~3.4E4932