

含原字符的阴影处理技巧

李红胜 (随州市第一人民医院)

摘要:本文介绍用 C 语言调用 DOS 系统资源,来实现含原字符的阴影的处理技巧。

一、引言

在我们所接触的一些商品软件中,有些屏幕处理相当漂亮,比如立体窗口,其立体窗口的黑色阴影并非是简单地填充黑色,而是含有阴影覆盖处的原来字符,这样的阴影比简单地填充要漂亮得多,立体感更强。

我们用 Foxbase 或 Clipper 设计应用软件时,由于系统本身的局限,所以无法达到含原字符阴影的处理效果,这使得很多程序设计人员感到遗憾。但是,如果我们利用 C 语言调用 DOS 系统资源,则可以轻而易举实现这一功能。

二、原理

我们知道,屏幕上要实现阴影的方法是:在需要开窗口的坐标上错位,设置阴影颜色后清屏,一般以黑色作为阴影。

那么,含原字符的阴影是怎样实现的呢?

屏幕上每个字符,在显示内存中是用两个字节来存储的,其中低位字节存放的是字符本身,高位字节存放的是字符的属性。如果能够获取阴影部分的字符,并且按阴影属性重新在屏幕写一遍,则含原字符的阴影即可实现了。

DOS 中 ROM-BIOS 的 10H 中断是视频输入、输出,当 AH 为 8 时,可以读取光标处的字符及属性;当 AH 为 9 时,可以向光标处写字符及属性。

三、方法

用 C 语言编写一个读光标处字符及属性的函数 sread(),再编写一个向光标处写字符及属性的函数 swrite(),以上两个函数供主程序 main()调用,其中坐标处理及光标移动在主程序 main()中完成。

具体请见所附源程序 SCHANG.C 及演示程序

SCH.PRGR,如在汉字状态下使用,则必须是支持直接写屏的汉字系统。在 FoxBASE 中的使用格式如下:

! SCHANGE TOP LEFT BOTT RIGHT

参数说明:TOP 和 LEFT 为阴影部分左上角坐标;BOTT 和 RIGHT 为阴影部分右下角坐标。

```
/* 如下程序用 Turbo C 编译运行成功. */
/* SCHANGE.C 源程序 */
#include "dos.h"
#include "stdlib.h"
#include "conio.h"
main(int argc, char * argv[])
{int left, top, right, bott, i, j;
 union scan {int c;
 char ch[2];
 }sc;
 if(argc < 5) exit(0);
 /* 坐标处理 */
 top = atoi(argv[1]+1);
 left = atoi(argv[2]+1);
 bott = atoi(argv[3]+1);
 right = atoi(argv[4]+1);
 for(i = left; i <= right; i++)
 for(j = top; j <= bott; j++)
 {gotoxy(i, j);
 sc.c = sread();
 swrite(sc, ch[0]);
 }
}

/* 读光标处字符及属性函数 */
sread()
{union REGS r;
 r.h.ah = 8;
 r.h.bh = 0;
 int86(0x10, &r, &r);
}

/* 向光标处写字符及属性函数 */
swrite(char zf)
{union REGS r;
 r.h.ah = 9;
 r.h.bh = 0;
 r.h.cl = 1;
 r.h.ch = 0;
 r.h.al = zf;
 r.h.bl = LIGHTGRAY BLACK * 16;
 int86(0x10, &r, &r);
}
```

参考文献:

[1] Herbert Schildt [美] 著, 郭兴社、戴建鹏编译《C 语言大全》电子工业出版社 1992