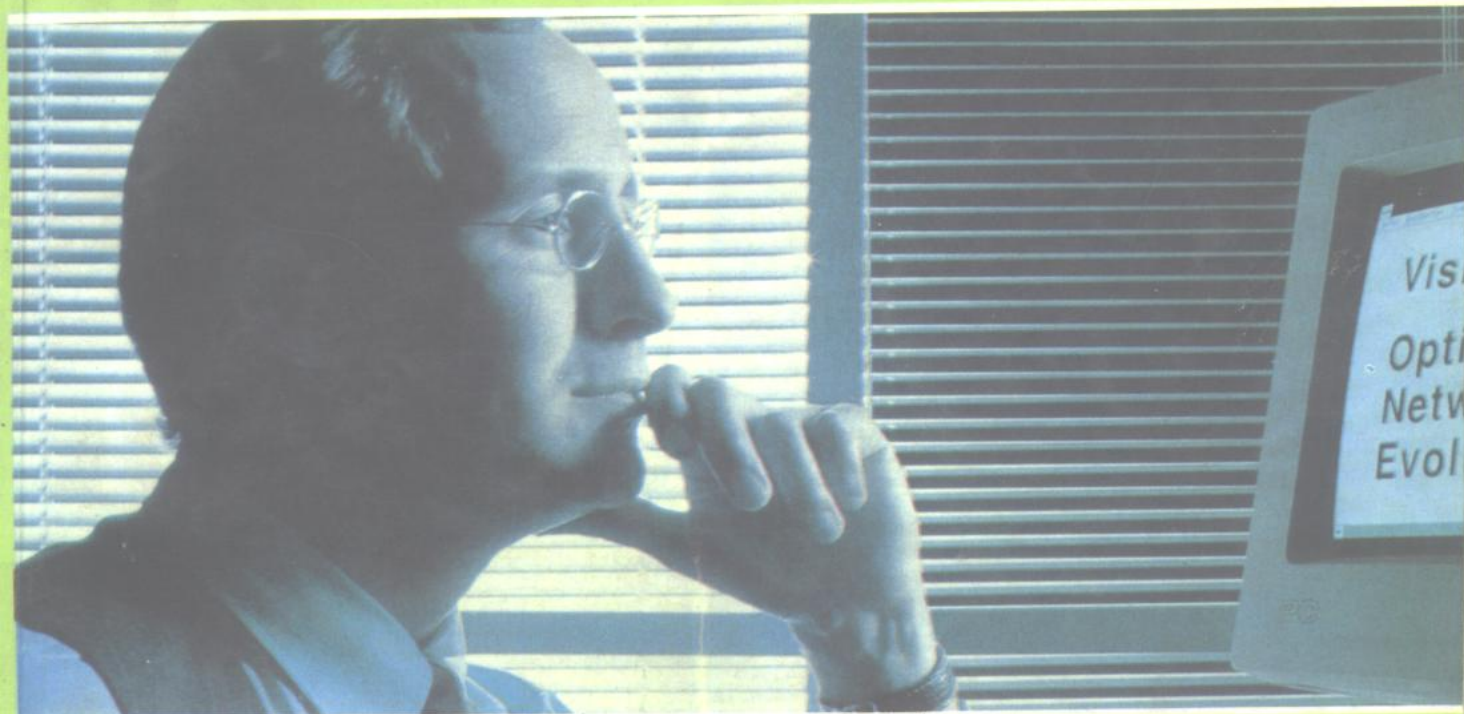


鲁沐浴 主编



语言编程技巧 及实用程序荟萃

科学出版社

2
1/1

C 语言编程技巧及实用程序荟萃

鲁沐浴 主编

科学出版社

1 9 9 4

(京)新登字 092 号

内 容 简 介

C 语言在国内广泛流行,深受广大计算机应用开发者的喜爱,其应用水平也在不断提高,但低水平的重复开发仍较普遍,耗费人力物力。为此,本书将众多 C 语言应用经验、应用示例、实用程序,经过筛选、提炼、分类后汇编成册,供广大计算机用户,特别是 C 语言应用开发者、爱好者学习和选用。

本书可以说是 C 语言应用技巧,解决实际问题的创造性、灵活性、实用性的集中反映,是应用开发者的结晶。本书每篇文章中,虽没有技术理论的全面论述,但针对性强,一事一论,短小精悍,实用价值大,在一般教课书中难以找到。

JS254/23

C 语言编程技巧及实用程序荟萃

鲁沐浴 主 编

责任编辑 徐津津

· 科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

中国人民解放军第一二〇二工厂印刷

新华书店北京发行所发行 各地新华书店经售

1994 年 7 月第一版 开本 787×1092 1/16

1994 年 7 月第一次印刷 印张 26

印数 1-5000 字数:600 000

ISBN 7-03-004207-7/TP·374

定价:22.00 元

目 录

第一部分 菜单

1.1	在 UNIX 操作系统下用 C 语言编制 ORACLE 菜单	1
1.2	用 Turbo C 实现具有立体投影效果的选择菜单	2
1.3	用 C 语言编制光条和弹出式菜单实例	3
1.4	用 C 语言实现光条菜单	5
1.5	用 C 语言实现汉字菜单的方法	6
1.6	彩色汉字菜单的实现方法	7
1.7	在汉字环境下实现弹出菜单	10
1.8	用 C 语言编写菜单程序	13
1.9	用 C 语言编写的菜单程序	16
1.10	在 XENIX 下光数菜单的实现	17
1.11	用 Turbo C 实现键控彩色菜单	19
1.12	Turbo C 实现彩色汉字菜单的几个关键点	20
1.13	用 MSC 5.0 实现菜单按钮程序	22
1.14	弹出式菜单中屏幕滚动的具体实现	23
1.15	用 VGA 卡剩余帧缓存开发屏幕菜单	26
1.16	Turbo C 菜单和 FoxBASE 连运技术	28
1.17	用 C 语言实现双字节制表符生成方框程序	30
1.18	在菜单程序中执行大型 DOS 程序的方法	31

第二部分 输入

2.1	C 语言对键盘处理功能的扩充方法	33
2.2	用 Turbo C 编制的键盘宏定义程序	34
2.3	用 C 语言设计数字化仪图形输入系统	37
2.4	模拟键盘输入通用子程序	39
2.5	窗口环境下带过滤的字符串输入函数	41
2.6	一种实现命令行编辑功能的方法	43
2.7	简单的图形态下交互式输入输出程序	47

2.8	键盘缓冲区直接存取例程.....	48
2.9	在 UC DOS 中挂上任意输入码的方法.....	50
2.10	文本方式下鼠标器编程的几个常用的 C 语言子程序	52
2.11	以鼠标器为热键的 TSR 程序的编制.....	55
2.12	支持鼠标操作的 BIOS INT 33H 中断使用方法与文本及图形方式下定义 鼠标器形状的演示程序	59

第三部分 屏幕显示

3.1	能实现文本文件阅读器功能的程序.....	65
3.2	优化的通用文本文件阅读器.....	68
3.3	EGA/VGA 图形模式下文本显示程序	70
3.4	实现文本状态下的阴影窗口程序.....	74
3.5	EGA 圆弧显示修正算法及圆弧子程序	76
3.6	CEGA、CVGA 中文显示卡扩展图形的编程方法	78
3.7	CGA 显示器低分辨率屏幕图形设置技术	82
3.8	长城 CVGA 卡 ZOOM 功能的一种特殊用途——窗口幻灯示范程序.....	85
3.9	IBM PC 机 VGA 显示拍摄技术	87
3.10	C 语言编程中的视频显示和用户界面	89
3.11	通用文件显示 TYPEE 程序	92
3.12	Turbo C 中时钟自动显示的实现	95
3.13	一个实用的文件阅读程序——FLRD·C	96
3.14	用 Turbo C 编写的 TYPE MORE PRINT 功能三合一的程序	98
3.15	Turbo C 中文显示技巧	100
3.16	用 Turbo C 语言显示汉字的技巧	103
3.17	实现立体菜单的一种简易方法.....	104
3.18	实用的 C 语言汉字显示输出方法	105
3.19	在 Turbo C2.0 图形方式下显示彩色汉字的方法	107
3.20	西文 DOS 下彩色汉字的快速显示	108
3.21	图形方式下实现汉字的无级缩放.....	111
3.22	MS DOS 下汉字可变颜色显示的方法	113
3.23	用 Turbo C 实现中西文字符串快速显示	115
3.24	西文状态下显示汉字的新方法.....	116
3.25	在西文状态下汉字特殊字型的显示程序.....	119
3.26	用 Turbo C 在长城机上编写的显示汉字串的函数	120

3.27	在文本方式下显示汉字的实例	121
3.28	中文彩色立体窗口的实现方法	122
3.29	分层式多窗口软件的设计与实现	123
3.30	用 Turbo C Tools 开发汉字菜单和窗口显示软件	128
3.31	在西文 Turbo C 中直接利用中断 INT10 实现 2.13H 的特殊显示功能	131
3.32	利用 Turbo C 的图形页技术实现动画显示	132
3.33	用 C 语言实现内存驻留程序(TSR)实例介绍	134
3.34	用 C 程序控制屏幕上下滚动和左右平移的方法	136
3.35	微机屏幕图象的压缩存储与恢复	139
3.36	彩色汉字屏幕的保存与恢复	141

第四部分 打印

4.1	用 C 实现 EGA/VGA 图形的存取与打印	143
4.2	从打印机上输出 VGA 高分辨率屏幕图形	145
4.3	VGA 高分辨率屏幕图形的汉字显示及图形打印	145
4.4	用 SGP.COM/SEGP.COM 打印图形的方法及例程	149
4.5	Turbo C 通用打印程序	151
4.6	用 Turbo C 编写的打印源程序清单的程序	152
4.7	“打印”屏幕至磁盘文件的程序	157
4.8	XENIX 系统下的终端打印技巧	158
4.9	一个通用终端打印程序	160
4.10	选择页号进行打印的 C 程序	163
4.11	条形统计的直接打印	166
4.12	一个文本文件打印程序	170
4.13	打印机控制码序列转换程序	172
4.14	在 ORACLE 中生成中文格式报表的三种方法	174

第五部分 图形和汉字处理

5.1	用 AutoCAD 进行作图的方法与技巧	177
5.2	实现 AutoCAD 图形共享的一种方法	179
5.3	AutoCAD 图形交换文件的 C 语言实现	182
5.4	在中文方式下 Turbo C 及其图形功能的使用	183
5.5	用 C 语言快速地给 AutoCAD 点变量赋值	185
5.6	PC 机图形适配器橡皮筋技术及程序实例	186

5.7	在C语言中利用随机函数实现动画技术实例	188
5.8	书写动画的一种制作方法	189
5.9	在Turbo C图形视区中输出待定字符串的办法	191
5.10	扩充Turbo C线型作图函数的输出模式	194
5.11	在DOS命令行下显示SPT图形的C程序	195
5.12	用C语言调用SPT图形一法	196
5.13	精确复制SPT图形的简单方法	197
5.14	两种图形文件互相转换程序	199
5.15	WPS系统中SPT图形的放大程序	203
5.16	财务统计数据图形化的实现方法	204
5.17	Turbo C++ 2.0屏幕图形写入磁盘文件及由磁盘文件写入屏幕的方法 ...	206
5.18	Turbo C程序设计中的汉字处理技术	207
5.19	在Turbo C2.0环境下的一种汉字处理方法	213
5.20	用Turbo C编写的中文图符编辑程序	214
5.21	README.COM的使用方法和汉化	215
5.22	汉化软件工具的设计与实现	218

第六部分 文件的查找、修改、删除和恢复

6.1	XENIX下C程序命令行参数的一个特殊处理	221
6.2	快速磁盘搜索程序	222
6.3	一个通用的C语言文件查找程序	223
6.4	文件查找实用程序	225
6.5	文件的全盘模糊查找程序	228
6.6	用Turbo C编写的增强DOS查找功能的程序	229
6.7	用Turbo C编写的增强显示和查询磁盘文件的实用程序	231
6.8	查询星期的一种方法	232
6.9	与DOS的DIR命令功能相似的C语言程序	233
6.10	用Turbo C编写的修改文件创建时间的程序	236
6.11	超级文件属性修改程序	237
6.12	用C语言编写的删除目录实用程序	238
6.13	全盘范围内删除指定的文件	240
6.14	Turbo C2.0“物理性”删除实用程序	241
6.15	物理上删除磁盘文件的另一方法	244
6.16	DOS目录级操作的实现——用以删除和拷贝子目录的程序	245

6.17	ORACLE 数据备份与恢复的方法	247
6.18	内存文本恢复简法.....	249
6.19	在 XENIX 系统下恢复数据的一种方法	249
6.20	FoxBASE 受损文件的修复方法	251
6.21	在 UNIX 下按文件类型或时间界限列目录程序	251
6.22	UNIX 系统下能完成 DEBUG 功能 D 的程序	253
6.23	在 DOS 环境下的 C 语言文件连接程序	254
6.24	用 Turbo C 2.0 编写的文件搬动命令(move)	255
6.25	DOS 状态下移动文件的几种方法	256
6.26	一组处理 DBF 文件的 C 语言核心函数	257
6.27	DEBUG 和改向功能结合所得文件的有关问题的处理方法	259
6.28	EXE2BIN 命令逆过程的实现方法及实用程序	261

第七部分 数据安全

7.1	在 XENIX 下用 C 语言恢复 Crypt 的功能	264
7.2	给 UNIX/XENIX 系统超级用户开个后门	265
7.3	为 UNIX/XENIX 系统添加一个简单的多重加密器	266
7.4	在 XENIX 系统下的 C 语言文件加密程序	267
7.5	在 UNIX 系统下的 C 语言文件加密程序	268
7.6	用 C 语言对数据库信息加密的两种方法	270
7.7	反跟踪函数编写一法	271
7.8	一个为 com 文件设置通行字的程序 comlock	272
7.9	一个通用的硬盘 exe 文件加锁程序	276
7.10	在 XENIX 中对用户文件起保护作用的一个程序	277
7.11	用 C 语言为 FoxBASE+数据库加密程序	278
7.12	FoxBASE 伪编译文件的反编译	279
7.13	C 语言实用文件保护程序.....	287
7.14	用 C 语言进行激光加密的方法	287
7.15	多用户环境下的文件加密方法.....	289
7.16	一种用 EXE 代替 BAT 文件的方法	291
7.17	加密 BAT 文件二法	291
7.18	为 DOS 操作系统增加用户注册登记功能	292
7.19	给 Turbo C 增加一个口令函数	296
7.20	找回 WPS 文件中被忘记的密码	297

7.21	WPS 文件的解密方法	297
7.22	WPS 加密系统的解密方法	298

第八部分 C 语言与其它语言的混合应用

8.1	在 UNIX 环境下用 C 语言调用汇编语言子程序的方法	300
8.2	Turbo C 语言与汇编语言混合编程探讨	301
8.3	C 语言和 BASIC 语言混合编程方法	304
8.4	dBASE III 的自动维修程序	307
8.5	DOS 与 XENIX 文本格式转换程序	309
8.6	正确用 C 语言读取数据库	310
8.7	对《正确用 C 语言读取数据库》一文的补充	312
8.8	用 C 语言实现 UNIFY 数据库间的数据共享	314
8.9	Turbo Pascal 与 Turbo C 动态数组的一种用法	315
8.10	C 语言调用 BIOS 的通用程序	316
8.11	如何在 C 语言中使用 BIOS 数据及实用程序	318
8.12	在 DOS 中直接访问数据库记录和结构的 C 程序	320

第九部分 其它

9.1	补遗 DOS 功能的 C 语言程序	324
9.2	UNIX 操作系统 TAR 命令的改进	325
9.3	用 Turbo C2.0 开发的检查 DOS 内存程序	326
9.4	能检出和分离 C 语言源程序错误的方法	329
9.5	用 C 语言编写中断服务程序	330
9.6	PC SHELL7.0 中.FNT 字型库的利用	332
9.7	让普通用户直接关闭 UNIX 系统的程序	334
9.8	用 C 语言编制的 FoxBIND 仿真程序	335
9.9	调用词组的 C 语言程序	338
9.10	源程序规范化处理软件的开发与例程	339
9.11	用 C 语言实现不同类型的数据在内存中的混合存储	341
9.12	完全清除内存程序的程序	342
9.13	用 C 语言编制类似 FoxBASE 中的 INKEY(n)函数	343
9.14	用 C 语言编写的光标管理函数	344
9.15	用 C 语言对金额数据进行处理的方法	345
9.16	数制转换程序	347

9.17	多用户方式下报表数据求和的一种编程方法·····	347
9.18	实用的源程序统计程序·····	351
9.19	用 Borland C++设计的类内递归程序 ·····	352
9.20	C++与面向对象的并程序程序设计方法 ·····	353
9.21	实现高随机度随机序列的一种方法·····	356
9.22	用 PC 机自身的定时功能进行精确定时 ·····	357
9.23	用 Turbo C++编写的去掉 Wordstar 分页符的程序 ·····	359
9.24	用 Turbo C 将 WS 文书文件转换成 TXT 文件的方法 ·····	359
9.25	对 WS 文件的一点处理 ·····	360
9.26	使用西文 WordStar 的几点注意事项 ·····	362
9.27	一个简化华光排版系统使用的程序·····	362
9.28	给西山中文系统图像文件加上用户自己的色板·····	366
9.29	自动添加段结束符 \$ ·····	371
9.30	用 huffman 编码技术压缩数据及压缩程序 ·····	372
9.31	为 FoxBASE 增加一实用程序——过程分解器 ·····	376
9.32	编译 FoxBASE 程序状态开关的变换机理及实现程序 ·····	378
9.33	在程序中访问扩展内存的方法·····	382
9.34	给 MS DOS5.0 增加一条外部命令·····	386
9.35	C 语言中数学错误的处理方法·····	387
9.36	对 Turbo C 中库函数 EXEC 错误的修改 ·····	388
9.37	C 语言编程中的一个小技巧·····	390
9.38	程序执行时间的几种测量方法·····	391
9.39	也谈程序执行时间的测量方法·····	392
9.40	在 C 语言中调用系统资源的方法及技巧 ·····	393
9.41	Microsoft C 内存资源的充分利用 ·····	396
9.42	C++语言应用中值得注意的几个问题·····	398
9.43	C 语言中指针使用的常见错误·····	401
9.44	也谈 C 语言指针应用中的问题 ·····	402
9.45	在 C 语言编程中易犯的错误 ·····	404

第一部分 菜单

1.1 在 UNIX 操作系统下用 C 语言编制 ORACLE 菜单

笔者在 Unix 操作系统下用 C 语言编了一段程序,编译后生成的执行文件较为简短,可作为调用 ORACLE 的 SQL * Forms、SQL * Plus 以及各管理子系统的总控菜单,各分级菜单再由 SQL * Forms 的触发子技术调用。这里主要介绍该总控模块。程序中语句 5 是根据终端的特性,进行清屏并把显示置为高亮度属性,6、7 二句对称使用,把标题设为倍高倍宽显示,改变了 ORACLE 格式中字体单一的形象,突出了总标题。以下各句类似。第 25 句中的...033[4A\033]30C...使光标定位于“请根据需要选择 0~9”之后等待输入。各子系统用 C 语言的开关语句进行分支,具体调用 ORACLE 的各部分则利用 C 语言中执行 Unix 系统命令 System(); 的功能,如语句 system(“runform /flform 用户名/口令”); 为调用物资供应子系统格式,以下各项均如同该句,进入格式后正常执行格式的各项功能,当退出后,通过转移语句 goto b; 又回到总菜单。本程序在 Unisys 6000/51B 小型机、ORACLE6.0 版本下通过运行,终端为长城 GW220,仿真 VT100。

程序如下:

```
#include "stdio.h"
main()
{
    int c,count;
    printf("\033[? 3l\033[1m\n\n\n\n");
    printf("\033#3 上海纺织轴承一厂\n");
    printf("\033#4\上海纺织轴承一厂\n");
```

```
printf("\n");
printf("\033#6 计算机辅助管理信息系统\n\n\n\n\n\n\n\n\n");
printf("\033#6\033[5m 上海纺织轴承一厂计算机室编制\n\n");
printf("\033#6\033[0m 1990 年\n\n\n\n\n");
for (count=1;count<=2000000;count++);/*
    延时语句 */
b:
printf("\033[? 3l\n\033[1m");
printf("\033#3 * * * 信息系统总菜单 * * * \n");
printf("\033#4 * * * 信息系统总菜单 * * * \n");
printf("\033#6 ————\n\n\n");
printf("\033#6 1. 物资供应系统      2. 劳资人事系统\n\n\n");
printf("\033#6 3. 技术工装系统      4. 生产控制系统\n\n\n");
printf("\033#6 5. 财务统计系统      6. 计量管理系统\n\n\n");
printf("\033#6 7. 设备能源系统      8. 调用 SQL * PLUS\n\n\n");
printf("\033#6 9. 调用 SQL * FORMS  0. 退出信息系统\n\n\n");
printf("\033#6 请根据需要选择 0~9\n\n\n");
printf("\033#6\033[3A\033[30C");
while ((c=getchar())!= '0')
switch(c)
{ case '1':
    printf("\033[? 3l\033[9;25H 正在调用 ORACLE 请稍等! \n");
```

```

system("runform flform user/passwd");
goto b;
case' 2' :
    printf("\033[? 3l\033[9;25H 正在调用
        ORACLE 请稍等! \n");
    system("runform f2form user/passwd");
    goto b;
case' 3' :
case' 4' :
case' 5' :
case' 6' :
case' 7' :
    printf("\033[? 3l\033[7;6H\033# 6
        本系统正在开发! \n");
    for (count = 1; count <= 2000000; count
        ++);
    goto b;

```

```

case' 8' :
    printf("\033[? 3l\033[9;24H 正在调
        用 SQL * FORMS 请稍等! \n");
    system("sqlforms user/passwd");
    goto b;
case' 9' :
    printf("\033[? 3l\033[9;24H 正在调
        用 SQL * PLUS 请稍等! \n");
    system("sqlplus user/passwd");
    goto b;
default:
    goto b;
}
printf("\033[? 3l"); /* 清屏语句 */
}

```

(袁焕民)

1.2 用 Turbo C 实现具有立体投影效果的选择菜单

本文拟就 2.13H 汉字操作系统为例,说明其构造方法。2.13H 有一个菜单选择文件 MENUHH.COM 是黑白的,现将其改为具有立体投影效果的选择菜单,同时增加了日期显示和设置。读者只需将源码输入并编译、连接生成 EXE 文件取代 AUTOEXEC.BAT 文件中的 MENUHH.COM 即可,程序清单附下。

本程序在 Turbo C2.0 下实现,在 PC 系列机上通过。

程序清单:

```

#include <dos.h>
#include <stdio.h>
#include <graphics.h>
#define LEFTX    15
#define UPY      5
#define RIGHTX   65
#define DOWNY    19
#define DLINE(X,Y,C) {gotoxy(X,Y);textattr
    (0x17);putch(C);}
main()

```

```

    struct date dat;
    int    i, ye, mo, da;
    char    c;
    union    REGS reg;
START:    /* 显示菜单边框 */
    window(1,1,80,25);
    DLINE(LEFTX,UPY,0xc9);
    for (i=LEFTX+1;i<RIGHTX;i++){
        DLINE(i,UPY,0xcd); DLINE(i,UPY+2,
            0xc4);}
    DLINE(RIGHTX,UPY,0xbb);
    for (i=UPY+1;i<DOWNY;i++) DLINE
        (RIGHTX,i,0xba);
    DLINE(RIGHTX,DOWNY,0xbc);
    for (i=RIGHTX-1;i>LEFTX;i--) DLINE
        (i,DOWNY,0xcd);
    DLINE(LEFTX,DOWNY,0xc8);
    for (i=DOWNY-1;i>UPY;i--) DLINE
        (LEFTX,i,0xba);
    window(LEFTX+1,UPY+1,RIGHTX-
        1,DOWNY-1);

```

```

clrscr(); textcolor(RED);
gotoxy(3,2);cputs("Welcome you!");
AGAIN:/* *** 显示系统日期 *** */
window(LEFTX+1,DOWNY-1,RIGHTX
-1,DOWNY-1);
textbackground(GREEN);clrscr();textcol-
or(BROWN);
getdate(&dat);gotoxy(12,1);
cprintf("Current's date is %2d. %02d. %
02d,dat.da-year,dat.da-mon,dat.da-day);
/* *** 选择 z 菜单窗口的投影窗口 ***
*/
textbackground(BLACK);window(27,
10,56,16);clrscr();
/* *** 显示菜单内容 *** */
window(26,9,55,15);textbackground
(CYAN);
textcolor(WHITE);clrscr();
cputs(" \n");
cputs(" CCBIOS2.13H \n");
cputs("1__HZK16 ON HARD DRIVE
\n");
cputs(" 2__DOS X.XX \n");
cputs(" 3__HZK16 ON VDISK \n");
cputs("CR__1 HALF TO MEMORY \n");

```

```

reg.h.ah=0x0c;
intdos(&reg,&reg);/* 清键盘缓冲区 */
gotoxy(26,6);c=getche();
window(LEFTX+1,DOWNY-1,
RIGHTX-1,DOWNY-1);
textcolor(WHITE);textbackground
(GREEN);clrscr();
if(c=0x1b)/* ESC 设定系统日期 */
{
gotoxy(11,1);
cputs("Enter new date is:");
cscanf("%4d.%02d.%02d",&ye,&mo,
&da);
dat.da__year=ye;
dat.da__mon=mo;
dat.da__day=da;
setdate(&dat);
goto Again;
}
gotoxy(20,1);cputs("Please wait
...");
bioskey(c);/* 返回键入字符的 ASCII */
}

```

(鲁红斌)

1.3 用 C 语言编制光条和弹出式菜单实例

利用 C 语言丰富的图形函数编制菜单,是一种较为通常的方法。

笔者认为这种方法有一些缺陷。C 的图形函数是为作图开发的,一般应在图形显示模式下使用,因此在有些汉字系统的显示模式中,不能正确运行。有些 C 语言版本将图形函数放在一个模块中,只要使用其中一个图形函数,整个模块就要链接到程序中,使执行程序增大。

如果程序只是作汉字菜单的话,不一定要使用 C 的图形函数。下面笔者结合一个程序实例,谈一种不用 C 图形函数编制菜单的方法。

程序分为二个模块:显示控制模块 menu.c 和 BIOS 调用模块 bcall.c。显示控制部分较简单,稍具 C 知识的读者不难看懂,在此不再作说明。

BIOS 调用中的函数功能说明如下:

1. LOCATE(int cur __1,...)一定光标的位置,cur __1=行光标,cur __c=列光标。
2. GET __KEY()—读按键值,该函数可读取功能键的扩展代码。
3. CHANGE __COLOR(int com ...)—改变指定屏幕块的颜色,块中的显示内容不变。本函数用 INT10H 的 8 号子功能读

取字符,改变属性后用 INT10H 的 9 号子功能写上屏幕,实现修改光条菜单的色彩,置彩色屏幕块和阴影效果。读者可对该函数作些修改,使之能直接显示彩色字符串。

参数说明如下:

com=0,改变块的色彩;

com=1,使指定块产生阴影;

x1,y1一块左上角行、列光标;

x2,y2一块右上角行、列光标;

color—彩色值,背景色 * 16 + 字符色。

本程序可在大多数汉字系统下运行,实现方法简单。由于篇幅限制,只能提供基本的功能,有兴趣的读者可自己扩充完善,相信读者一定能从中得到帮助。

编译方法:本程序用 MICROSOFT C5.0 以上版本编译。如将函数形参说明移出,也可以用 MICROSOFT C5.0 以下版本编译。

CL/c *.c / * 生成 menu.obj, bcall.obj * /

LINK menu+bcall; / * 产生 menu.exe 文件 * /

本程序在 AST386、北极星 286 等机上运行通过。汉字系统为金山 DOS 和 JW = DOS。

/* Filename is bcall.c */

#include <dos.h>

#include <io.h>

union REGS regs;

LOCATE (int cur __1, int cur __c)

```
{
    regs.x.ax=0x0200;
    regs.x.dx=(cur __1*8)+cur __c;
    int86(0x01, &regs, &regs);
}
```

GET-KEY()

```
{
    regs.h.ah=0;
    int86(0x16, &regs, &regs);
    if(regs.h.al==0) return(regs.h.ah+128);
    else return(regs.h.al);
}
```

CHANGE __COLOR(int com, int cur __x1, int cur __y1, int cur __x2, int cur __y2, int color)

```
{
    int y __1, asc __ii, cur __p[4]; cur __y1-=1;
    cur __x1-=1;
    if(com!=0)
    {
        cur __p[0]=cur __x1; cur __p[1]=cur __y1;
        cur __p[2]=cur __x2; cur __p[3]=cur __y2;
        cur __x1=cur __x2; cur __y1+=2; cur __x2+=1;
        cur __y2+=2;
        if(color==0) color=0x08;
    }
    start: while(++cur __x1<=cur __x2)
    {
        y __1=cur __y1;
        while(++y __1<=cur __y2)
        {
            regs.x.ax=0x0200;
            regs.x.dx=(cur __x1*8)+y __1;
            int86(0x10, &regs, &regs);
            regs.h.ah=0x03;
            int86(0x10, &regs, &regs);
            asc __ii=regs.h.al;
            regs.x.cx=0x0001;
            regs.h.bl=color;
            regs.h.al=asc __ii;
            regs.h.ah=0x09;
            int86(0x10, &regs, &regs);
        }
    }
    if(com!=0)
    {
        cur __x1=cur __p[0]+1; cur __y1=cur __p[3];
        cur __x2=cur __p[2]; cur __y2=cur __p[3]+2;
        com=0; goto start;
    }
    /* Filename is menu.c */
    #include <dos.h>
    int select='A', in __cur=10;
    main()
    {
        int i;
```



```

if(c=='¥015&&j!=LEN-1) zx(j+1);
switch(c)
{
case'j':
printf("¥033[%d;%dh/¥033[0m%s",j+2,
COL,a[j+1]);
j++;
if(j==LEN) j=0;
break;
case'1':
zx(1);break;
case'2':
zx(2);break;
case'3':

```

```

zx(3);break;
case'0':
printf("/n/033[0m");exit(1);
}
}
zx(j)
int j;
{
char comm[20];
sprintf(comm,"menu%d",j);system(comm);
}

```

(何炳新)

1.5 用 C 语言实现汉字菜单的方法

本文介绍用 MS Quick C 在汉字操作系统 UC DOS2.01 下实现汉字窗口菜单的具体方法及一些应注意的问题,并列出了一些程序的简略结构。

当弹出式或下拉式菜单被激活后,它就覆盖当前屏幕菜单区的显示内容。菜单消失后,屏幕又恢复原来的状态。这从用户的角度看,屏幕上显示的信息量增大了,有时几个菜单可以同时显示,而用户的注意力不会受到影响。

一般西文菜单工作在文本方式下,而中文显示则需将屏幕设置成图形方式。故此,设计汉字菜单就有许多不同的地方。下面就设计中文菜单应注意的几个问题加以探讨。本文所有程序均用 MS Quick C 编成,并在 UC DOS2.01 汉字操作系统下运行通过。

1. 显示器设置

UC DOS2.01 支持 25 行汉字显示,每个汉字显示在屏幕上占 8×14 点阵,X 方向为 8 点,Y 方向为 14 点。UC DOS 系统配置可以将屏幕显示器设置为 EGA 或 VGA。在进行中文菜单设计时,笔者确定将显示器设置为 EGA640×350:

```

setvideomode(_ERESCOLOR);/*
EGA640×350*/

```

这样,每行可显示 80 个汉字(80×8=640),一屏可显示 25 行(25×14=350)。如果将显示器设置成 VGA640×480,则会遇到一些麻烦。

2. 窗口菜单的弹出和消失

在窗口菜单被激活和显示之前,应事先把将要被覆盖的屏幕内容存入内存缓冲区,以备恢复屏幕时取用。程序采用动态分配内存的方法,及时申请缓冲区。缓冲区容量必须足够大(但不得大于 64KB),以保存从屏幕读取的信息。

菜单选择完毕,窗口菜单将消失,恢复屏幕只不过是上面缓冲区保存的信息放回到原来的位置。屏幕复原后,缓冲区即可释放掉,以节省内存。实现此过程的部分程序如下:

```

char far * buffer; /* 指向缓冲区的指针 */
buffer=(char far *)_fmalloc((unsigned
(int)_imagesize(x1, y1, x2, y2)); /* 动态分配内存缓冲区 */
_getimage(x1, y1, x2, y2,buffer); /* 保存屏幕信息到缓冲区 */
..... /* 窗口菜单操作 */

```

```
__putimage(x1, y1, buffer, __GPSET); /* 恢复
    屏幕背景 */
```

```
__free((char far *) buffer); /* 释放指针 */
```

利用这种方法,可以实现二级、三级子菜单等,道理同上。

3. 扫描码预定义

当菜单出现后,如果按上、下、左、右键,就会看到菜单或光标在随之移动。为了编程时表达明显、方便,常将这些方向键和其它一些常用键的扫描码进行预定义:

```
#define LEFT    75
#define RIGHT   77
#define UP      72
#define DOWN    80
#define ENTER   28
#define ESC     1
```

4. 光标的显示、移动和选择

菜单出现后,当前光标(也有人称之为“光棒”)位置的内容常用反显,以示醒目。这是如何实现的呢?简而言之,光标是“画”上去的,而汉字(或英文)菜单项是“写”上去的。

在同一个位置,写了汉字然后画矩形光标,或者画了矩形光标然后再写汉字,这样后者都会将前者覆盖。为了实现在同一位置同时显示光标和文字,就得先用光标颜色画一矩形,并将其暂时存入缓冲区(方法同前),再把文字写上去,最后再把矩形块按一定的逻辑关系放到文字位置。移动光标时,需两次完成上述过程,其一将原菜单光标恢复成背景色,其二将下一项菜单置为前景色。这样看起来的效果就是光标移动,而实际是重复“画”和“写”的过程。在286微机上,这个过程是很快,根本感觉不到闪烁现象。

如果想选择当前光标位置的菜单,则只需按一下回车(ENTER)键。再加上一些并不

复杂的判断和计算,即可实现光标的循环移动。下面列出部分程序的简略结构框架:

```
switch (((_bios _keybrd(_KEYBRB
    ____READ)&0xff00)>>8)) /* 键盘扫描
    */
{
    case RIGHT: ..... /* 右移 */
    case LEFT:  ..... /* 左移 */
    case UP:    ..... /* 上移 */
    case DOWN:  ..... /* 下移 */
    case ENTER: ..... /* 选中 */
    case ESC:   .....
    default:    .....
}
```

5. 阴影的实现

窗口菜单带阴影,增强了立体感,效果很好。一种比较简单的方法是,在窗口的右边和下边分别加一个颜色较深的矩形长条。这样,阴影下面的背景将被覆盖。要想同时显示背景内容,则可以通过一定的逻辑运算获得满意的效果。

6. 改善屏幕显示

UCDOS 汉字系统的显示功能,是用新的 INT10H 代替原有的 INT10H,可以利用该功能来改变光标,使自动光标显示或不显示。如果显示自动光标,则会破坏画面美观。当菜单移动时,屏幕上会留下自动光标的痕迹(一条小细线)。这一点,正好可以利用UCDOS 的系统接口,通过修改中断而使之得到改善。部分汇编程序如下:

```
mov ah, 1
mov ch, 0b0h
int 10h
.....
```

(乔爱科)

1.6 彩色汉字菜单的实现方法