

C程序绘图

程 进

(四川省工业贸易学校,四川 成都 610081)

【摘 要】C语言是目前世界上流行,使用最广泛的、面向过程的高级程序设计语言,许多大型应用软件都是用C语言编写的,C是函数式的语言,利用标准库函数和自己设计的函数可以完成许多功能。C语言具有绘图能力强、可移植性,并且具备很强的数据处理能力,因此适于编写系统软件、三维、二维图形和动画,它是数值计算的高级语言,本文以较为常用的TURBOC2为例介绍如何利用图形处理函数绘图问题。

【关键词】C程序;函数绘图;图形处理

【中图分类号】TP319.41 **【文献标识码】**A **【文章编号】**1673-1891(2009)02-0051-04

众所周知,C语言是函数式的语言,各种不同版本的C语言提供了十分丰富的库函数,C语言也提供了较强的图形处理函数,现以较为常用的TURBOC2为例介绍如何利用图形处理函数绘图。

1 图形模式的初始化

在使用图形函数编程之前,要先进行初始化处理,即将字符模式转化为图形模式。PC机有多种控制显示的图形卡,TURBOC2一般支持8种常用的图形卡,它们分别是:

彩色图形卡 CGA

多色图形阵列卡 MCGA

增强图形适配器 EGA,EGA64,EGAMONO

IBM8514 适配器

视频图形阵列 VGA

大力神图形适配器 HERCMONO

AT&T400线图形适配器 ATT400

3270PC图形适配器 PC3270

不同的图形卡有不同的显示模式,在每一种显示模式下具有不同的分辨率和颜色值。所谓分辨率就是将屏幕划分为许多点,例如分辨率为 320×200 表示将屏幕划分为 320×200 个点,每一个点称为一个像素,每个像素有一个坐标,左上角为(0,0),右下角坐标为(319,199)。分辨率越高,则图形越清晰。TURBOC支持的图形显示模式如表1所示:

表1 TURBOC支持的图形显示模式

图形卡	图形模式	分辨率	颜色数
CGA	CGA0,CGA1,CGA2,CGA3	320×200	4
	CGAHI	640×200	2
	MCGA0,MCGA1,MCGA2,MCGA3	320×200	4
MCGA	MCGAMED	640×200	2
	MCGAHI	640×480	2
EGA	EGALO	640×480	16
	EGAHI	640×200	16
EGA64	EGA64LO	640×200	16
	EGA64HI	640×350	4
	EGAMONO	EGAMONOH	640×350
HERCMONO	HERCMONOH	720×348	2
	ATT400C0,ATT400C1,ATT400C2,ATT400C3	320×200	4
ATT400	ATT400MED	640×200	2
	ATT400HI	640×400	2
	VGALO	640×200	16
VGA	VGAMED	640×350	16
	VGAHI	640×480	16
PC3270	PC3270HI	720×350	2

收稿日期:2009-04-24

作者简介:程进(1958-),男,讲师,主要从事机械原理及机械设计的教学与研究。

TURBBOC的图形初始化由initgraph函数完成,使用该函数应包含graphics.h头文件。该函数的功能是选定图形驱动程序及显示模式,一般格式为: initgraph(&gd,&gm,""),其中gd为驱动程序,gm为显示模式。目前大多数微机的驱动程序为VGA,显示模式为VGAHI,因此,初始化一般格式可这样设置:

```
gd=VGA;gm=VGAHI;initgraph(&gd,&gm,"")。
```

也可自动检测驱动程序和显示模式,下面的小程序就能自动检测:

```
#include<graphics.h>
main()
{
int gd,gm,x,a,b,p,k,s;
gd=DETECT; /*自动检测*/
initgraph(&gd,&gm,"");
x=getmaxcolor()+1; /*获取最大颜色数*/
```

```
a=getmaxx(); /*获取最大X座标*/
b=getmaxy(); /*获取最大Y座标*/
k=getdrivername(); /*获取图形驱动程序名*/
s=getmodename(gm); /*获取图形模式*/
p=getgraphmode(); /*获取图形模式值*/
closegraph();
printf("maxcolor:%d\n",x);
printf("maxx:%d\n",a);
printf("maxy:%d\n",d);
printf("graphdriver:%-20s%d\n",k,gd);
printf("graphmode:%-20s%d\n",k,s,p);
getch();
}
```

2 颜色设置

在绘图时要设置颜色,目前大多数微机提供16种颜色,其颜色及代号如表2所示:

表2 绘图颜色及代号

颜色	代号	颜色	代号	颜色	代号	颜色	代号
黑色	0	红色	4	深灰	8	淡红	12
蓝色	1	紫红色	5	浅蓝	9	浅紫	13
绿色	2	褐色	6	浅绿	10	黄色	14
青色	3	浅灰	7	淡青	11	白色	15

颜色设置通常要设置前景色和背景色,前景色由setcolor()函数设置,背景色由setbkcolor()函数设置。例如:setcolor(1)表示前景色为蓝色,setbkcolor(4)表示背景色为红色。

前景色一般不能和背景色相同,如相同则图形不可见,也可用这种方式实现动画显示。下面的程序就是将各种颜色及代号显示在屏幕上。

```
#include <stdlib.h>
#include <graphics.h>
main()
{
int I,gm,gd;
char a [1]
gd=DETECT;
initgraph(&gd,&gm,"");
for(i=1;I <=15;i++)
{
itoa(i,a,10; /*数字转化为字符*/
setcolor(i);
rectangle(20+40*(i-1),0,60+40*(i-1),479); /*画矩形*/
setfillstyle(1,1);
```

```
floodfill(40+40*(i-1),240,i);
setcolor(i+2);
settextstyle(3,0,2);
outtextxy(40+40*(i-1),240,a);
}
getch();
}
```

3 填充模式设置

在图形绘制时,有时需要对一块封闭区域填充,就需要设置填充模式,用setfillstyle函数设置,其格式为:setfillstyle(pattern,color),color为填充的颜色,pattern为填充模式,不同模式见表3:

利用floodfill()函数填充,例如floodfill(x,y,z)表示用当前的填充模式填充以(x,y)为起点的被颜色为z封闭的区域。下面的程序就是以不同模式填充一个太极图。

```
#include <dos.h>
#include <graphics.h>
main()
{
int gdriver=VGA,gmode=VGAHI,i;
initgraph(&gdriver,&gmode,"") /*初始化*/
```

表 3 绘图填充模式

代号	模式	代号	模式	代号	模式	代号	模式
0	背景色填充	3	左斜线填充	6	右斜线填充	9	交叉线填充
1	单色填充	4	粗左斜线填充	7	浅影填充	10	松空白填充
2	横线填充	5	粗右斜线填充	8	淡交叉线填充	11	紧空白点填充

```

for(i=0;i<=12;i++)
{
cleardevice();/*清屏*/
setcolor(1);/*前景色*/
arc(320,240,0,180,200);/*画弧*/
arc(420,240,180,360,100);
arc(220,240,0,180,100);
setfillstyle(i,1);/*设置模式*/
floodfill(320,200,1);/*填充*/
setcolor(4);
arc(420,240,180,360,100);
arc(220,240,0,180,100);
arc(320,240,180,360,200);
setfillstyle(i,4);
floodfill(320,280,4);
circle(420,240,35);
floodfill(420,240,4);
setcolor(1);
circle(220,240,35);/*画圆*/
setfillstyle(i,1);
floodfill(220,240,1);
sleep(1);/*暂停 1 秒*/
}
getch();/*等待输入*/
closegraph();/*关闭图形模式*/
}
    
```

```

initgraph(&gdriver,&gmode,"");
pcircle();
prectangle();
pling();
pbar();
}
pcircle()
{
cleardevice();
while(!kbhit())/*当无键击下时循环*/
{
setcolor(random(14)+1);/*随机设置前景颜色*/
circle(random(640),random(480),random(60));/*画
随机圆*/
}
getch();
}
prectangle()
{
cleardevice();
while(!kbhit())
{
setcolor(random(14)+1);
rectangle(random(640),random(480),random(640),
random(480));
}
getch();
}
pline();
{
cleardevice();
while(!kbhit())
{
setcolor(random(14)+1);
line(random(640),random(480),320,240);/*画直线
*/
}
getch();
}
pbar()
    
```

4 常用的一些画简单图形的函数

bar(x1,y1,x2,y2)功能是用当前填充模式和填充颜色画一填充长方形,长方形顶点坐标为(x1,y1)和(x2,y2)。circle(x,y,r)功能是以(x,y)为圆心 r 为半径画圆。line(x1,y1,x2,y2)功能是画一直线,顶点为(x1,y1)和(x2,y2)。Rectangle(x1,y1,x2,y2)功能是画一矩形,其对角线顶点为(x1,y1)和(x2,y2)。

下面的程序的功能就是产生各种图形。

```

#include <stdlib.h>
#include <stdio.h>
#include <graphics.h>
main()
{
int gdriver=VGA,gmode=VGAHI,
    
```

```

{
int a,b,c,d;
cleardevice();
while(!kbhit())
{
setfillstyle(random(12),random(14)+1);
a=random(640);
b=random(480);
c=random(640);
d=random(480);
bar(a,b,c,d);/*画填充矩形*/
}
getch();
}

```

5 图形模式下的文本输出

在字符模式下文本的输出常常采用 printf 函数，在图形模式下要用 settextstyle 函数设置文本字形、显示方向和字符的大小。其格式为 settextstyle(font, direction, charsize), font 为字形设置参数, 其值和对应的字形如下: 0 8 × 8 位图字形; 1 三倍笔划字形; 2 小号笔划字形; 3 无衬线笔划字形; 4 黑体笔划字形。

derection 为显示方向参数, 它有两个值, 值为 0 时表示从左向右显示, 值为 1 时表示从下向上显示。

Charsize 为字符放大因子。其值为 0 表示字符放大因子为 4, 即字符显示在 4 × 4 像素矩形区域内; 其值为 1 表示放大因子为 8, 即字符显示在 8 × 8 像素矩形区域内; 其值为 3 表示字符显示在 16 × 16 像素矩形区域内; 依此类推, 最大到普通大小的 10 倍。

字符显示模式设置好后, 就可用 outtextxy 函数将文本输出来, 其格式为 outtextxy(x,y,str), 参数 x 和 y 表示文本输出的起始坐标, str 为输出的字符串。

注释及参考文献:

[1] 谭浩强.C 程序设计[M].北京:清华大学出版社,1991.
[2] 谭浩强.C 程序设计题解[M].北京:清华大学出版社,1992.
[3] 吕涛, 吕凤翥.C 程序设计实用问答[M].北京:北京大学出版社,1997.

下面的小程序的功能是将字符串 ABCDEFG 以不同的字形和大小按水平方式显示。

```

#include <graphics.h>
main()
{
int GraphDriver,GraphMode,h,i;
GraphDriver=DETECT;
initgraph(&GraphDriver,&GraphMode,"");
for(h=0;h <=4;h++)
{
for(i=1;i <10;i++)
{
setcolor(15);
settextstyle(h,0,i);/*设置文本模式*/
outtextxy(100,200,"ABCDEFGG");/*输出文本*/
sleep(1);/*暂停1秒钟*/

setcolor(0);
orttextxy(100,200,"ABCDEFGG");
}
}
setcolor(4);
settextstyle(3,0,10);
orttextxy(100,200,"OVER!");
getch();
closegraph();
}

```

6 其它函数介绍

cleardevice()的功能是清除屏幕。closegraph()的功能是关闭图形模式, 回到文本模式。(注: 所有程序都在 PC 机上运行通过)

Function Cartography with C Procedure

CHENG Jin

(Industry-trade School of Sichuan, Chengdu, Sichuan 610081)

Abstract: C is one of the most popular and widely used programming languages. The philosophy of c is that it is an imperative (procedural) systems implementation language. Although C was designed for implementing system software, it is also widely used for developing portable application software. Much large-scale application software is written with C. In C, all executable code is contained within functions by accessing standard C library function and using self-defined functions to accomplish many functions. Due to combination of desirable (下转 59 页)

(Department of Electronic Engineering, Yingkou Vocational and Technical College, Yingkou, Liaoning 115000)

Abstract: This paper puts forward one kind of teaching forum that 2000 data bases making use of ASP technology union SQL Server structure, that a forum based on B/S structure, structure of Yu WEB server, and to assist combining with the FTP technology for assist teaching, and discusses systematic construction and its technology and method .

Key words: Campus net; Education Forum; Assist teaching system

(上接35页)

(College of Territorial Resources and Tourism, Anhui Normal University, Wuhu, Anhui 241000)

Abstract: ArcGIS Engine is an embedded type component library newly launched by ArcGIS9. This paper introduces the technologies of the ArcGIS Engine and. NET, using C# language and ArcGIS Engine components to develop a special thematic map rendering system, and expound design flow for the system and some of the main functions of the implementation.

Key words: ArcGIS engine; C#.NET; Thematic map rendering

(上接54页)

characteristics such as code portability, strong graphic design capacity and data processing, C is suitable implementing system software, three and two dimensional graphic design, and the animation. It is the higher order language of value computation. This article takes more commonly used TURBOC2 as the example to introduce how to process function cartography question by using graphic processing.

Key words: C procedure; Function cartography; Graph processing