

FoxBASE 统计分析图形系统的设计

朱世平 (中国矿业大学)

摘要: 本文主要论述了在微机数据库管理系统 FoxBASE 下进行图形设计的一般方法。介绍了如何直接读取数据库文件, 进行必要的处理, 然后生成相应的图形, 并给出了一个具体实例。

一、系统设计的一般方法

由于 FoxBASE 不提供生成图形的命令, 因此, FoxBASE 统计分析图形系统的设计, 必须依靠具有图形功能的高级语言。在统计分析图形系统中应通过友好的用户界面, 让使用者对需要进行的分析或统计进行选择, 回答与图形输出的有关问题, 然后系统就能生成所需要的图形。FoxBASE 统计分析图形系统的设计主要应解决以下几个主要问题:

1. 必要的图形数据和图形画面信息

这部分数据根据统计分析图形系统和通用性和专用性要求, 在编程处理方式上有所不同。

对于通用系统, 它应让使用者回答, 哪些数据库文件参加处理, 处理的字段是哪些, 有什么条件, 是否需要某种分析或预测, 图形标题是什么, 图形副标题是什么, 需要输出哪种图形, 坐标单位等。但要用户回答这么多问题, 存在一定难度, 因此在许多场合, 可根据具体情况进行专用系统设计。

专用系统可以只让用户回答极少问题, 比如, 时间(哪一年至哪一年), 就能输出必要的图形, 至于哪些数据库文件参加处理, 哪些字段参加处理, 需要哪些画面信息, 完全由程序自动控制, 它是由用户与编程人员事先商量好的。

2. 图形数据的获得方法

只有获得图形数据, 才能进行图形输出。由于必要的数据库文件可能分散在各自的数据库文件中, 因此图形数据的获得有时是比较麻烦的, 既涉及某一数据库文件的计数与求和, 又涉及数据库文件之间的联系, 有时还要进行某些复杂的数学运算, 比如, 回归分析等。获得方法主要

有两种, 一种是利用 FoxBASE 提供的命令进行计数, 求和, 汇总, 然后把所得到的数据放入某一数据库文件中, 并经过适当格式转换供高级语言调用作图; 另一种就是使具有图形功能的高级语言能够直接对 FoxBASE 数据库文件进行操作, 提取必要字段的数据, 经过所需的数学运算, 得到有关的图形数据。

二、系统的具体实现

1. 接口设计

无论是直接从汇总的数据文件提取数据, 还是直接用高级语言读取有关的数据库文件进行处理, 都涉及到高级语言与数据库文件的数据交换问题。对于前一种情况可以采用 FoxBASE 的文本输出文件(.TXT)与高级语言交换数据, 但多少有些麻烦, 对于后一种情况就必须解决对数据库文件的直接操作, 当掌握了 FoxBASE 数据库文件和内部结构之后, 我们就可以找到这种方法。

FoxBASE 库文件由结构区和数据区两部分组成, 而结构区又分库文件整体描述和库字段描述两部分。库文整体描述部分占 32 个字节, 该 32 个字节一般只有前几个字节有用。

库字段描述信息由数据库中字段的定义信息构成, 每个字段有一个长 32 字节的字段说明, 因此库字段描述信息的总长度为字段数 \times 32。

另外, 在字段描述信息后面, 紧跟着一个字节是数据库文件结构块结束字节, 在 FoxBASE 中该字节为 ODH。

因此结构区总长度的计算方法是:

$$32 \times \text{字段个数} + 33$$

结构区后就是数据区, 即数据库记录内容。数据记

录由一个空格(20H)字节开始,然后是该记录中各字段内容,字段内容均以 ASCII 码表示,字段内容按其定义的顺序依次排列,相互之间没有分隔符。在所有记录之后,是库文件的结束标志,在 FoxBASE 中,它是十六进制数 1AH。

在了解了数据库文件内部结构之后,使用高级语言来操纵数据库就不难了。我们知道 C 语言把文件看成是一个字符的序列,即由一个一个字符(字节)的数据顺序组成,而不考虑记录的界限。在 Turbo BASIC 中也有这种文件形式。有了这种文件形式,我们就可以方便地用高级语言对 FoxBASE 的数据库文件进行直接操作。在读取数据时,只要将文件指针跳过结构区开始读就行了。比如有一个 FoxBASE 的数据库文件 student.DBF,其结构如下:

字段名	类型	宽度	小数
姓名	字符型	10	
年龄	数字型	2	
性别	字符	2	

该数据库文件有 10 个记录,我们可以用 C 语言编写一个直接读取该文件数据的小程序。

```
#include <stdlib.h>
#include <stdio.h>
struct student Type
{
char de[1];
char Name[10];
Char age[2];
Char sex[2];
} stud [10];
Main()
{
int i;
FILE * fp;
If ((fp=fopen("student.DBF","rb"))=Null)
{
printf("cannot open file \n");
exit(o);
}
```

```
fseek (fp,129l,o);
for (i=0; i<10; i+t)
{fread (& stud[i], sizeof(struct student-Type),
1,fp);
printf("%S\n", stad[i], Name);
}
fclose (fp);
}
```

2. 图形生成

目前最广泛使用的显示适配器有:MDA, CGA, EGA, VGA。长城 0520CH 的彩色板比较特殊,在其上输出图形,需要使用其特有的虚拟图形设备。有关这些适配器的显示方式可在有关的资料上找到。

在屏幕上作图,首先要检测出是种适配器,然后正确地转换到某种图形方式,这一点在 Microsoft C 中可通过 -getvideoconfig 函数和 -setvideomode 函数很容易作到。另外机器隐含的屏幕坐标,即原始坐标系,其左上角为坐标原点,如图 1 所示;

这与数学中通常使用和笛卡尔坐标系差距很大,如果按照原始坐标系设计图形画面,显得很不容易,所以应该充分利用高级语言中对原始坐标重新定义的命令。



图 1

了解了具体的屏幕象素情况,然后根据确定的实用坐标系,设计图形画面就变得容易了。

下面我们给出一个具体的实例。在高等学校科研经费的管理系统中,可以按图 2 进行总体设计。

其中在图形输出部分,需要对各个系部某一年所获科研经费的百分比做出统计,并且希望以扇形图方式输出,我们可以用数据库文件 KYJfInqk.dBF 放入汇总后的信息,其结构如下:

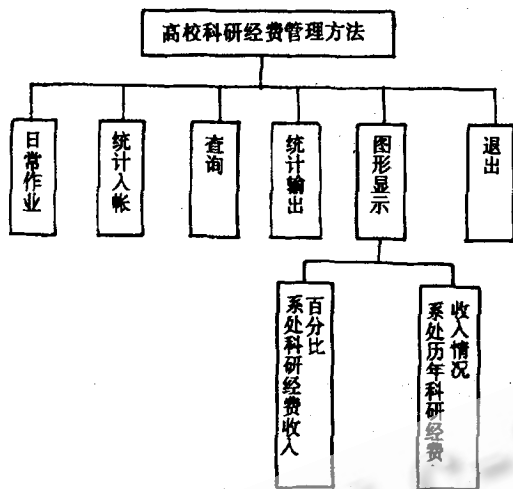


图 2

在这个文件当中,经过程序的处理,放入了各系历年科研经费收入情况。

当我们选择科研经费管理系统的图形输出模块中“系处科研经费收入百分比”后,只要回答屏幕上的年代提问,就会在屏幕上输出相应的图形,如图 3 所示(程序清单附后)

字段名	字段类型	字段长度	小数位
(系名)unibm	c	8	
(科研经费收入)unibmsr	n	8	3
(年时间)unitime	c	2	
(系英文简名)unibmeng	c	4	

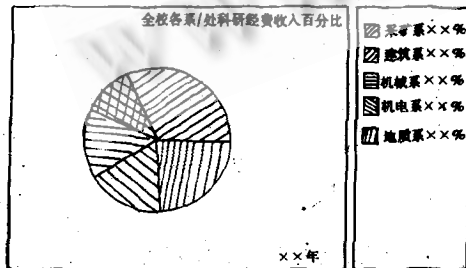


图 3

同时我们还可以利用 KYJfjqk.dbf 文件,以直方图的形式输出某系历年科研经费收入情况如图 4 所示。(程序省略)

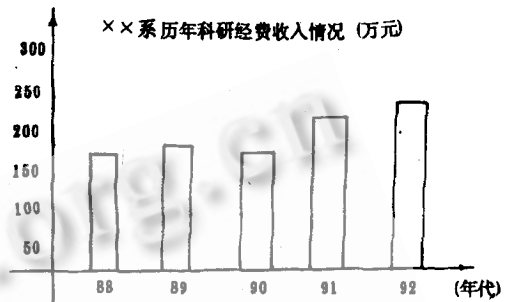


图 4

此外我们还可以利用 KYJfjqk.dbf 文件,采用某种预测方法,如灰色预测,对下一年学校科研经费收入情况作出预测。

总之,经过这样一些设计,使整个系统更加生动和完善,从实际结果看,其方法也是完全可行的。

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <graph.h>
#define SIZE 21
char u;
struct uni
{
char unistr[8];
}t1[SIZE],u1[SIZE],t2[SIZE],u2[SIZE];
struct unii
{
char unyee[2];
}t3[SIZE],u3[SIZE];
struct uniii
{
char unieng[4];
}t4[SIZE],u4[SIZE];
struct uniiii
{
char copyy[8];
}c1[1];
    
```

```

main()
{
    int i,i1,j,ix,iy,ix1,iy1,le,M3;
    double M, M1, M2, x, y, tat;
    short colo, j1, y1;
    char year[2];
    FILE * fp;
    if ((fp = fopen("kyjflnqk.dbf", "rb")) == NULL)
    {
        printf("cannot open file\n");
        exit(0);
    }
    clrscr( GCLEARSCREEN);
    settxtwindow(10, 30, 11,50);
    printf("%s", "input year:");
    scanf("%s, &year);
    i = 1;
    i1=0;
    fseek(fp,161L,0);
    while (!feof (fp))
    {
        fread (&u,1,1,fp); /* read delete unit */
        fread(& t1[i], sizeof (struct uni), 1, fp); /* read field
1 * /
        fread(& t2[i], sizeof (struct uni), 1, fp); /* read field
2 * /
        fread(& t3[i], sizeof (struct uni), 1, fp); /* read field
3 * /
        fread(& t4[i], sizeof (struct uni), 1, fp); /* read field
4 * /
        if (strncmp (t3[i]. unye, year, 2) == 0)
        {
            strcpy(u1[i]. unisr, t1[i]. unisr);
            strcpy(u2[i]. unisr, t2[i]. unisr);
            strcpy(u3[i]. unye, t3[i]. unye);
            strcpy(u4[i]. unieng, t4[i]. unieng);
            i1 = i1+1;
        }
        i++;
    }
    fclose(fp);
    clrscr( GCLEARSCREEN);
    setvideomode( VRES16COLOR);
    settxtwindow(2,15,3,50);
    outtext("全校各系 / 处科研经费收入百分比");
    rectangle(2,2,2,430,430);
    rectangle(2,440,2,639,430);
    tat = 0;
    M1 = 0;
    j1 = 2;
    y1 = 21;
    colo = 0;
    iX1 = 360;
    iy1 = 210;
    for(j= 1;j<i1;j++)
    {
        M = atof(u2[j]. unisr);
        tat = tat+atof(u2[j]. unisr);
    }
    for(j= 1;j<i1;j++)
    {
        M = atof(u2[j]. unisr);
        M2 = atof(u2[j]. unisr) / ata * 100; /* percent */
        M = atof(u2[j]. unisr) / tat * 360 * 0.0174532925;
        M1 = M1+M;
        x = 210+150 * cos(M1);
        y = 210-150 * sin(M1);
        ix = (int)x;
        iy = (int)y;
        if(j = 1)
        {
            ix1 = 360;
            iy1 = 210;
        }
        pie( GFILLINTERIOR, 60, 60, 360, 360, ix1,iy1
        ix, iy);
        rectangle(3, 460, y1, 470, y1+10);
        settxtwindow(j1, 62, j1+1, 70);
        strncpy(c1[1]. copyy,u1[j]. unisr, 6);
        outtext(c1[1].copyy);
        settxtwindow(j1, 70, j1+1, 79);
        printf("%f%%%\n", M2);
        j1 = j1+1;
        y1 = y1+18;
        colo = colo+1;
        setcolor(colo);
        ix1 = ix;
        iy1 = iy;
    }
    settxtwindow(23,40,24,50);
    outtest(year);
    settxtwindow(23,44,24,50);
    outtext("年");
    settxtwindow(23,70,24,79);
    outtext("百分比%");
    getch();
    setvideomode( TEXTC80);
}

```