**C语言直接操作FORTRAN数据文件的方法**

张云如

　　**摘　要：**FORTRAN数据文件的结构及用C语言直接对FORTRAN数据文件进行数据读取和处理的方法，并结合实例，给出了完成操作的C语言源程序。
　　**关键词：**C语言　FORTRAN数据文件　数据直接读写

　　FORTRAN语言适用于数值计算，在科学和工程计算方面发挥了重要作用。C语言具有很强的通用程序设计功能，而且很适于处理图形图像、系统程序方面的问题，但它的数值计算能力不如FORTRAN语言。如果能在C语言程序中直接调用FORTRAN程序的计算结果，则可充分利用这2种语言的优势。为此，有必要建立用C语言对FORTRAN数据文件进行直接读取和处理的方法。

**1　FORTRAN数据文件的结构**
　　在C语言程序中调用FORTRAN程序的计算结果，可通过FORTRAN数据文件来实现，而要用C语言直接存取FORTRAN数据文件，必须首先清楚FORTRAN数据文件的结构。
　　FORTRAN数据文件按存取方式的不同可分为顺序文件和直接文件。而不论是顺序文件还是直接文件，按数据在文件中的存放方式的不同，都有2种存放形式，一种是字符形式(即ASCII码形式)，另一种是二进制代码形式。前者称为有格式存放，后者称为无格式存放。由于对直接文件存取的基本操作步骤与对顺序文件存取的基本操作步骤相同，二者的不同主要体现在操作时是否应顺序进行。下面以顺序文件为例进行具体分析。
**1.1　有格式文件的结构**
　　有格式文件全部由有格式的记录组成，有格式的记录是字符的序列，以ASCII码形式存放。各记录以回车符(0DH)和换行符(0AH)作为结束标志，各记录之间无分隔符，各记录的长度(字节数)可以不等。
**1.2　无格式文件的结构**
　　无格式文件全部由无格式记录组成，无格式记录是值的序列，即二进制代码序列。这种格式的文件，其第1个字节是4BH(即十进制的75)，最后1个字节为82H(即十进制的130)，各无格式记录的数据信息位于这2个字节之间。
　　每一个无格式记录都由下面3部分组成：(1)记录的首字节，是该记录所有数据所占的字节数；(2)组成该记录的各数据内容。不论是整型数据还是实型数据，每个数据均占4个字节，其中整型数据只占4个字节中的前2个字节(低字节在前)，而后面的2个字节为空(20H)。各数据之间不空格。(3)每个无格式记录的末字节，又是该记录的字节数，而不是以回车换行符作为结束标志。各记录之间也没有分隔符。

**2　用C语言直接操作FORTRAN数据文件的方法**
**2.1　直接读取FORTRAN数据文件**
　　由于有格式文件是ASCII文件，因而用C语言读取它时，只要用“r”方式打开它，便可逐字符读取该文件中的数据，直至遇到文件结束标志为止。
　　用C语言读取无格式文件时，一般先用“rb”方式打开它，然后用fseek函数将文件指针定位到第2个字节处，先读取第1个无格式记录的字节数(它一定为4的倍数)，然后自第3个字节开始，每次用fread函数(不能用fscanf函数)读取4个字节，便可将该无格式记录中的数据逐个读出，直至达到第1个无格式记录的字节数。若要读取第2个无格式记录中的诸数据，可在读完第1个记录之后(此时，文件指针已定位于第1个字节的末字节处)，将文件指针再前移1个字节，先从该字节读出第2个记录的字节数，之后的操作便与读取第1个记录的相同。其余记录的读取可依此类推。
　　在创建FORTRAN无格式文件时，如果将整数写成同值的实数，则在用fread函数读取时，可免去进行所读数据是整数还是实数的判断，而又不影响数据值的大小。
**2.2　直接更新FORTRAN数据文件中的记录**
　　对于有格式文件，可先以“r+”方式打开它，然后将文件指针定位到要更新记录的开始字节处，再用fprintf函数(不能用fwrite函数)写上新的数据内容，由于有格式记录以回车换行符结束，因此，当写上的新内容不足原来的长度时，应以空格填满至回车符(0DH)之前。
　　对于无格式文件，可先以“rb+”方式打开它，然后将文件指针定位到要更新记录的开始字节之后的下1个字节，再用fwrite函数(不能用fprintf函数)，每4个字节写上新的数据内容，最后，将文件指针分别定位到该记录的开始字节以及末字节，修正该记录的字节数。
**2.3　直接增删FORTRAN数据文件中的记录**
　　对于有格式文件，可按下列步骤添加记录：(1)以“r+”方式打开它；(2)将文件指针定位于要插入的位置；(3)将文件指针所指位置之后的所有记录用fscanf函数读入内存；(4)重新将文件指针定位于要插入的位置；(5)把要插入的新记录内容用fprintf函数写到文件中；(6)将已读入到内存的所有记录再用fprintf函数写回到文件中(于新记录之后)。
　　对于无格式文件，添加记录的步骤同上，但应将打开方式改成“rb+”，读文件记录用fread函数，写文件记录用fwrite函数，并注意无格式记录无回车换行符这一结束标志。
　　不论有格式文件还是无格式文件，删除某记录的步骤都是：先将要删除记录之后的所有记录读入内存，然后将文件指针定位于要删除记录的开始处，最后将已读到内存的那些记录重新写回到文件中。

**3　举例**　　下面通过2个实例说明用C语言程序直接操作FORTRAN数据文件的实现方法。在调用例1的程序之前，先用FORTRAN语言生成1个FORTRAN有格式顺序文件。在调用例2的程序之前，先用FORTRAN语言生成1个FORTRAN无格式顺序文件。为使程序能读取任意1个FORTRAN数据文件，这里将main函数处理成带参数的形式，在调用程序时，只需在DOS命令行键入执行程序名和相应的FORTRAN数据文件名。可将程序运行结果与对应的FORTRAN数据文件进行比较。2个程序均在Turbo C2.0下通过。
　　例1：此程序的功能是读取1个FORTRAN有格式顺序文件，并将其在屏幕上显示。它与DOS的TYPE命令功能相当。
　　#include “stdio.h”
　　main(int argc,char \*argv［］)
　　{char ch,fname［13］;
　　　FILE \*fp;
　　　if(argc<2)
　　　　{printf(“＼nInput file name:”);
　　　　scanf(“%s”,fname);
　　　　}
　　　else
　　　　strcpy(fname,argv［1］)；
　　　fp=fopen(fname,“r”);
　　　if(fp==NULL){
　　　　printf(“open %s error”,fname);
　　　　getchar();
　　　　exit(0);}
　　　printf(“Filename:%s＼n”,fname);
　　　while((ch=fgetc(fp))!=EOF)
　　　　printf(“%c”,ch);
　　　printf(“＼n”);
　　　fclose(fp);
　　}

　　例2：此程序先读取并屏显1个FORTRAN无格式顺序文件的某记录数据，然后用键入的数据更新该记录。
　　#include“stdio.h”
　　main(int argc,char \*argv［］)
　　{char t,fname［13］；
　　　int i; float x;
　　　FILE \*fp;
　　　if(argc<2){
　　　　printf(“＼nInput file name:”);
　　　　scanf(“%s”,fname);}
　　　else
　　　　strcpy(fname,argv［1］)；
　　　fp=fopen(fname,“rb+”);
　　　if(fp==NULL){
　　　　printf(“open %s error”,fname);
　　　　getchar();
　　　　exit(0);}
　　　printf(“Filename:%s＼n”,fname);
　　　fseek(fp,1L,0);
　　　fscanf(fp,“%c”,&t);　/\*读取第1个记录的字节数\*/
　　　printf(“%d＼n”,t);
　　　for(i=1;i<=t/4;i++)　/\*读取并屏显第1个记录
各数据\*/
　　　{fread(&x,4,1,fp);
　　　　printf(“%f”,x);
　　　}
　　　printf(“＼n”);
　　　scanf(“%d”,&t);
　　　fseek(fp,2L,0);/\*文件指针定位于首记录的首数据始\*/
　　　for(i=1;i<=t;i++)
　　　{scanf(“%f”,&x);
　　　　fwrite(&x,4,1,fp);　/\*更新首记录数据并回写到文
件中\*/
　　　}
　　　fputc(t\*4,fp);　/\*修改首记录的起止字节内容\*/
　　　fseek(fp,1L,0);
　　　fputc(t\*4,fp);
　　　fclose(fp);
　　}

**作者单位:**南昌大学计算中心(330029)

**参考文献**1　周大庆.C语言基础教程.长沙：中南工业出版社，1996
　2　谭浩强，田淑清.FORTRAN语言—FORTRAN77结构化程序设计.北京：清华大学出版社，1990