

FITSIO C++ 接口库libfio使用说明

v.1.0 2007/Jun/7th

顾俊骅

1 功能描述

libfio是一个cfitsio软件包在C++语言上的包装。通过libfio可以在C++语言中方便地操纵fits格式的文件，主要是fits图像。隐藏了cfitsio库中所要求的C语言指针语法，通过libfio可以大大地提高语言效率，精简程序长度。

2 安装

libfio需要用到cfitsio和blitz++两个库，

1.安装blitz++库:

```
>tar xvfz blitz-0.9.tar.gz
>cd blitz-0.9
>./configure
>make
>su (切换到root)
>(输入密码)
>make install
```

2.设置fitsio

若机器上已经安装了heasoft，则不用安装新的fitsio只需作如下的设置即可:

找到机器上fitsio.h和所在的位置，该文件通常位于heasoft/Linux_2.4.i686/include下

```
>cd /usr/include/
>ln -s <where fitsio.h is>/fitsio.h
>ln -s <where cfitsio.h is>/cfitsio.h
找到机器上libcfitsio.a和libcfitsio.so，该文件通常位于heasoft/Linux_2.4.i686/lib下
>cd /usr/lib
>ln -s <where libcfitsio.a is>/libcfitsio.a
>ln -s <where libcfitsio.so is>/libcfitsio.so
```

3.安装libfio包，进入刚才解压之后生成的fio目录下，注意，不是先前进行make的blitz目录

```
>make clean
```

```
>make
```

进行编译然后以超级用户执行以下命令:

```
>cp *.h /usr/include/
```

```
>cp libfio.so /usr/lib/
```

即安装完成。

3 libfio库的使用说明

3.1 头文件的包含

要使用libfio，在每个C++源程序文件中都要包含<fio.h>文件，通常而言，还需要写
using namespace blitz

3.2 编译选项

如果某个使用了libfio库的程序文件名为test.cc，那么可以用以下命令进行编译：
g++ test.cc -I/usr/local/include -L/usr/local/lib -lfio -lcfitsio

3.3 程序的编写

以下是一个实际的程序，通过对这个程序的分析，来解释一个libfio库的典型调用语法。

```
#include<fio.h> //包含fio头文件
#include<iostream> //包含C++输入输出库
#include<cmath> //包含标准库数学头文件

using namespace std;
using namespace blitz;
//主程序开始
int main(int argc, char* argv[])
//argc和argv用于获得命令行参数
//argc是命令行参数的个数
//argv[0], argv[1], argv[2], ..., argv[argc-1] 是各个参数字符串
//例如: ./a.out xxx yyy zzz
//则argc=4, argv[0]="./a.out", argv[1]="xxx", argv[2]="yyy", argv[3]="zzz"
{
    cfitsfile ff1; //声明一个用于操纵fits文件的对象。
    ff1.open("a.fits"); //打开一个已经存在的fits文件。
    Array<double, 2> img1; //声明一个用于存放图像的矩阵。
    ff1>>img1; //将数据从fits文件导入到矩阵中。

    //这里可以对这个矩阵中的数据进行任意的操纵。
    //假定这里计算输入图像的泊松误差:
```

```

Array<double,2> img2(img1.shape());
//注意C/C++的数组指标是从0开始
for(int i=0;i<img2.extent(0);++i)
{
    for(int j=0;j<img2.extent(1);++j)
    {
        img2(i,j)=sqrt(img1(i,j));
    }
}

cfitsfile ff2;          //声明另一个用于操纵fits文件的对象。
ff2.create("b.fits");//创建一个fits文件，如果已经存在，就先删除，然后
再创建。
ff2<<img2;              //将数据从矩阵导出到fits文件中。
}
//程序结束
//EOF

```

以下是另一个实例，这个实例读入一个fits图像文件，然后将其数据写入一个文本文件。

```

#include <iostream>
#include <cmath>
#include <cstdlib>
#include<utility>
#include <fstream>
#include "fio.h"
#include <cassert>
#include <vector>

using namespace std;

using namespace ::blitz;

int main(int argc,char* argv[])
{
    //检查用户提供的参数是否符合要求

    if(argc!=3)
    {
        cerr<<argv[0]<<" <input fits file> <output text file>"<<endl;
    }
}

```

```

        return -1;
    }
    //定义一个fitsfile对象，用于读入fits图像文件
    cfitsfile ff;
    //打开文件，文件名就是用户提供的第一个参数

    ff.open(argv[1]);

    //定义一个二维矩阵，名字为img
    Array<double,2> img;
    //将数据从fits文件读入img
    ff>>img;

    //定义一个对象用于操作磁盘上的文本文件
    ofstream ofs(argv[2]);

    //首先将图像文件的高和宽写入文本文件
    ofs<<img.extent(0)<<"\t"<<img.extent(1)<<endl;;

    //遍历所有的矩阵元素，将其依次写入fits文件
    for(int i=0;i!=img.extent(0);++i)
    {
        for(int j=0;j!=img.extent(1);++j)
        {
            ofs<<img(i,j)<<" ";
        }
        ofs<<endl;
    }
    //done
}

```