

# C++语言基础

迂者 - 贺利坚

<http://blog.csdn.net/sxhelijian/>

<http://edu.csdn.net>





本节主题:

文本文件(ASCII文件)的读写

# 打开磁盘文件

打开文件是指在文件读写之前做必要的准备工作

- (1) 为文件流对象和指定的磁盘文件建立关联，以便使文件流流向指定的磁盘文件
- (2) 指定文件的工作方式

打开文件的两种不同方法

- (1) 调用文件流的成员函数open，如

ofstream outfile;//定义输出文件流类对象

outfile.open("f1.dat",ios::out);

- (2) 在定义文件流对象时调用文件流类的构造函数

istream infile("c:\\new\\f1.dat", ios::in);



## 打开文件的方式

**ios::in** : 供读, 文件不存在时, 打开失败(ifstream默认的打开方式)

**ios::out** : 供写, 文件不存在则创建, 若文件已存在, 则清空原内容(ofstream默认的打开方式)

**ios::ate** : 文件打开时, 指针在文件最后; 可改变指针的位置, 常和in、out联合使用

**ios::app** : 供写, 文件不存在则创建, 若文件已存在, 则在原文件内容后添加写入新的内容

**ios::trunc** : 在读写前先将文件长度截断为0 (默认)

**ios::nocreate** : 文件不存在时产生错误, 常和in或app联合使用

**ios::noreplace** : 文件存在时产生错误, 常和out联合使用

**ios::binary** : 二进制格式文件

```
outfile.open("f1.dat",ios::out); //以 ios::out的方式打开文件
```

```
ofstream f("d:\\file.txt"); //默认以 ios::out的方式打开文件
```

```
ifstream f("salary.txt",ios::nocreate); //默认以 ios::in 的方式打开文件, 文件不存在时操作失败
```

```
fstream f("d:\\stu\\score.dat",ios::in|ios::out|ios::binary);//以读写方式打开二进制文件
```

```
ifstream fin("welcome.exe", ios::nocreate|ios::binary);
```

## 打开操作失败时.....



- ❏ 如果打开操作失败，open函数的返回值为0(假)，.....
- ❏ 如果是用调用构造函数的方式打开文件的，则流对象的值为0，.....
- ❏ 常用的处理方式形如

```
if(!outfile.open("fa.dat",ios::app))
{
    cerr<<"open error!";
    exit(1); // #include<cstdlib>
}
```

```
if(outfile.open("fa.dat",ios::app)==0)
    cerr<<"open error!";
else
    ... ..
```

```
if(outfile.open("fa.dat",ios::app))
{
    //打开成功的操作
}
```

```
ofstream outfile("f1.dat",ios::out);
if(!outfile)
{
    cerr<<"open error!"<<endl;
    exit(1);
}
```

# 关闭文件

- 在对已打开的磁盘文件的读写操作完成后，应关闭该文件。

```
int main( )  
{  
    int a[10];  
    ofstream outfile("f1.dat",ios::out); //打开  
    //处理逻辑，可以向文件中输出  
    .....  
    outfile.close(); //关闭磁盘文件"f1.dat"  
    return 0;  
}
```



- 关闭文件，将缓冲区中的数据妥善处理，解除了磁盘文件与文件流的关联，不再通过文件流对该文件进行输入或输出

## 示例：将数据写入ASCII文件

从键盘输入10个整数，并将其保存到数据文件f1.dat中。

```
#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;
int main( )
```

```
{
    int a[10];
    //打开文件
```

```
//操作文件
```

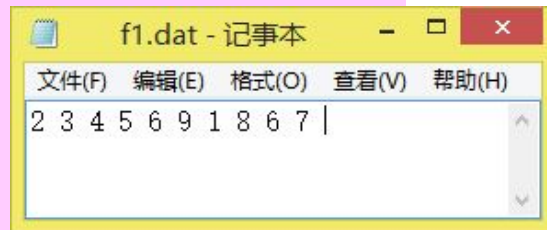
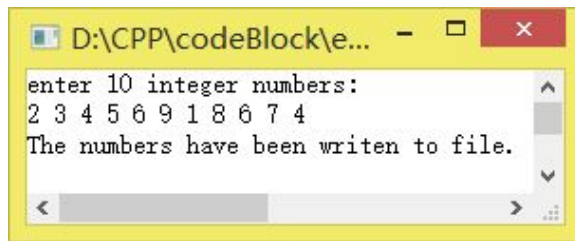
```
//关闭文件
return 0;
```

```
}
```

```
ofstream outfile("f1.dat",ios::out);
if(!outfile)
{
    cerr<<"open error!"<<endl;
    exit(1);
}
```

```
cout<<"enter 10 integer numbers:"<<endl;
for(int i=0; i<10; i++) //向磁盘文件"f1.dat"输出数据
{
    cin>>a[i];
    outfile<<a[i]<<" ";
}
cout<<"The numbers have been written to file. "<<endl;
```

```
outfile.close(); //关闭磁盘文件"f1.dat"
```



## 示例：从ASCII文件读入数据

从数据文件f1.dat中读入10个整数，找出并输出10个数中的最大者和它的序号。

```
ifstream infile("f1.dat",ios::in);
if(!infile)
{
    cerr<<"open error!"<<endl;
    exit(1);
}
```

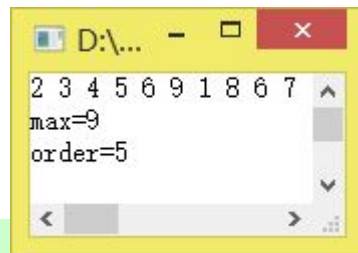
```
for(i=0; i<10; i++)
{
    infile>>a[i];
    cout<<a[i]<<" ";
}
cout<<endl;
```

```
infile.close();
```

```
#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;
int main( )
{
    int a[10],max,i,order;

    //打开文件
    //操作文件
    //关闭文件
    //其他工作
    return 0;
}
```

```
max=a[0];
order=0;
for(i=1; i<10; i++)
{
    if(a[i]>max)
    {
        max=a[i];
        order=i;
    }
}
cout<<"max="<<max<<endl;
cout<<"order="<<order<<endl;
```





# 对ASCII文件的操作

- ❏ ASCII文件，或称字符文件：文件中的每一个字节均以ASCII代码形式存放，即一个字节存放一个字符。
- ❏ 程序可以从ASCII文件中读入若干个字符，也可以向它输出一些字符。
- ❏ 对ASCII文件的读写操作的方法：

(1)用流插入运算符“<<”和流提取运算符“>>”输入输出标准类型的数据。

(2)用文件流的put, get, getline等成员函数进行字符的输入输出。

```
infile>>a[i];  
cout<<a[i];
```

```
cin>>a[i];  
outfile<<a[i];
```



## 示例：读写ASCII文件

```
void get_from_file()
{
    char ch;
    ifstream infile("f2.dat",ios::in);
    if(!infile)
    {
        cerr<<"open f2.dat error!"<<endl;
        exit(1);
    }
    ofstream outfile("f3.dat");
    if(!outfile)
    {
        cerr<<"open f3.dat error!"<<endl;
        exit(1);
    }
    while(infile.get(ch))
        outfile.put(ch-32);
    infile.close( );
    outfile.close();
}
```

从键盘读入一行字符，把其中的小写字母字符依次存放在磁盘文件f2.dat中。再把它从磁盘文件读出程序，将其中的小写字母改为大写字母，再存入磁盘文件

```
void save_to_file( );
void get_from_file();
int main( )
{
    save_to_file( );
    get_from_file( );
    return 0;
}
```

```
void save_to_file( )
{
    ofstream outfile("f2.dat");
    if(!outfile)
    {
        cerr<<"open f2.dat error!"<<endl;
        exit(1);
    }
    char c[80];
    cin.getline(c,80);
    for(int i=0; c[i]!=0; i++)
        if(c[i]>='a' && c[i]<='z')
            outfile.put(c[i]);
    outfile.close();
}
```

## 示例：在显示器上输出文件

```
#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;
void display_file(char *filename);

int main( )
{
    display_file("f3.dat");
    return 0;
}
```

```
void display_file(char *filename)
{
    ifstream infile(filename,ios::in);
    if(!infile)
    {
        cerr<<"open error!"<<endl;
        exit(1);
    }
    char ch;
    while(infile.get(ch))
        cout.put(ch);
    cout<<endl;
    infile.close();
}
```

# THANKS

本课程由 迂者-贺利坚 提供

CSDN网站：[www.csdn.net](http://www.csdn.net)  
企业服务：<http://ems.csdn.net/>  
人才服务：<http://job.csdn.net/>  
CTO俱乐部：<http://cto.csdn.net/>  
高校俱乐部：<http://student.csdn.net/>  
程序员杂志：<http://programmer.csdn.net/>

CODE平台：<https://code.csdn.net/>  
项目外包：<http://www.csto.com/>  
CSDN博客：<http://blog.csdn.net/>  
CSDN论坛：<http://bbs.csdn.net/>  
CSDN下载：<http://download.csdn.net/>

