

第7章 结构体和文件

第79课 文件操作

《信息学奥赛一本通·编程启蒙 C++版》

一、重定向版

【命令格式】

```
FILE * freopen ( const char * filename, const char * mode, FILE *  
stream );
```

【参数说明】

filename: 要打开的文件名

mode: 文件打开的模式, 和 fopen 中的模式 (r/w) 相同

stream: 文件指针, 通常使用标准流文件 (stdin/stdout/stderr)

其中 stdin 是标准输入流, 默认为键盘; stdout 是标准输出流, 默认为屏幕; stderr 是标准错误流, 一般把屏幕设为默认。通过调用 freopen, 就可以修改标准流文件的默认值, 实现重定向。

代码模版:

```
1. #include<cstdio> //使用 freopen 语句, 须调用 cstdio 库
2. int main()
3. {
4.     freopen("slyar.in", "r", stdin);
5.     freopen("slyar.out", "w", stdout);
6.     /* 中间按原样写代码, 什么都不用修改 */
7.     fclose(stdin);fclose(stdout);
8.     return 0;
9. }
```

【例 79.1】 文件操作

【题目描述】

从 `in.txt` 文件中读入数据，把它们之和保存 `out.txt` 文件中。

`in.txt` 数据：

1 2 3 4 5

`out.txt` 结果：

15

【代码实现】

```
1. #include<cstdio>
2. int main()
3. {
4.     freopen("in.txt","r",stdin); //定义输入文件名
5.     freopen("out.txt","w",stdout); //定义输出文件名
6.     int temp,sum=0;
7.     while (scanf("%d",&temp)==1) //(cin>>temp)从输入文件中读入数据
8.         //在C++中非0为真
9.     {
10.         sum=sum+temp;
11.     }
12.     printf("%d\n",sum); // cout<<sum<<endl;
13.     fclose(stdin);fclose(stdout); //关闭文件,可省略
14.     return 0;
15. }
```

二、fopen 版

重定向用起来很方便，但并不是所有算法竞赛都允许读写文件。甚至有的竞赛允许访问文件，但不允许使用 freopen 这样的重定向方式读写文件，可以使用 fopen 版，对 scanf 和 printf 语句适用。程序如下：

```
1. #include<cstdio>
2. using namespace std;
3. int main()
4. {
5.     FILE *fin,*fout;
6.     fin = fopen("in.txt","rb"); //定义输入文件名
7.     fout = fopen("out.txt","wb");//定义输出文件名
8.     int temp,sum=0;
9.     while (fscanf(fin,"%d",&temp)==1) //从输入文件中读入数据
10.    {
11.        sum=sum+temp;
12.    }
13.    fprintf(fout,"%d\n",sum);// cout<<sum<<endl;
14.    fclose(fin);fclose(fout); //关闭文件，可省略
15.    return 0;
16.}
```

先声明变量 fin 和 fout(暂且不用管 FILE *为何物)，把 scanf 改成 fscanf，第一个参数为 fin;把 printf 改成 fprintf，第一个参数为 fout，最后执行 fclose 关闭两个文件。

三、条件编译

一般情况下，C 语言源程序中的每一行代码都要参加编译。但有时候出于对程序代码优化的考虑，希望只对其中一部分内容进行编译，此时就需要在程序中加入条件，让编译器只对满足条件的代码进行编译，将不满足条件的代码舍弃，这就是条件编译（conditional compile）。

常见的条件编译指令

条件编译指令	说明
#if	如果条件为真，则执行相应操作
#elif	如果前面条件为假，而该条件为真，则执行相应操作
#else	如果前面条件均为假，则执行相应操作
#endif	结束相应的条件编译指令
#ifdef	如果该宏已定义，则执行相应操作
#ifndef	如果该宏没有定义，则执行相应操作

四、文件输入输出流

在 C++ 中，文件输入流 (ifstream) 和文件输出流 (ofstream) 的类，它们的默认输入输出设备都是磁盘文件。C++ 可以在创建对象时，设定输入或输出到哪个文件。由于这些类的定义是在 fstream 中进行的，因此，在使用这此类进行输入输出操作时，必须要在程序的首部利用 #include 指令包进 fstream 头文件。

例如：若想用 fin 作为输入对象，fout 作为输出对象，则可以使用如下定义：

```
ifstream  fin("输入文件名.扩展名");  
ofstream  fout("输出文件名.扩展名");
```

程序如下：

```
1. #include<fstream>//使用文件输入输出流，对 cin、cout 语句适用  
2. using namespace std;  
3. int main()  
4. {  
5.     ifstream fin("in.txt");//定义输入文件名  
6.     ofstream fout("out.txt"); //定义输出文件名  
7.     int temp,sum=0;  
8.     while (fin>>temp)  sum=sum+temp;//从输入文件中读入数据  
9.     fout<<sum<<endl;  
10.    fin.close();fout.close(); //关闭文件，可省略  
11.    return 0;  
12.}
```

谢谢!

—