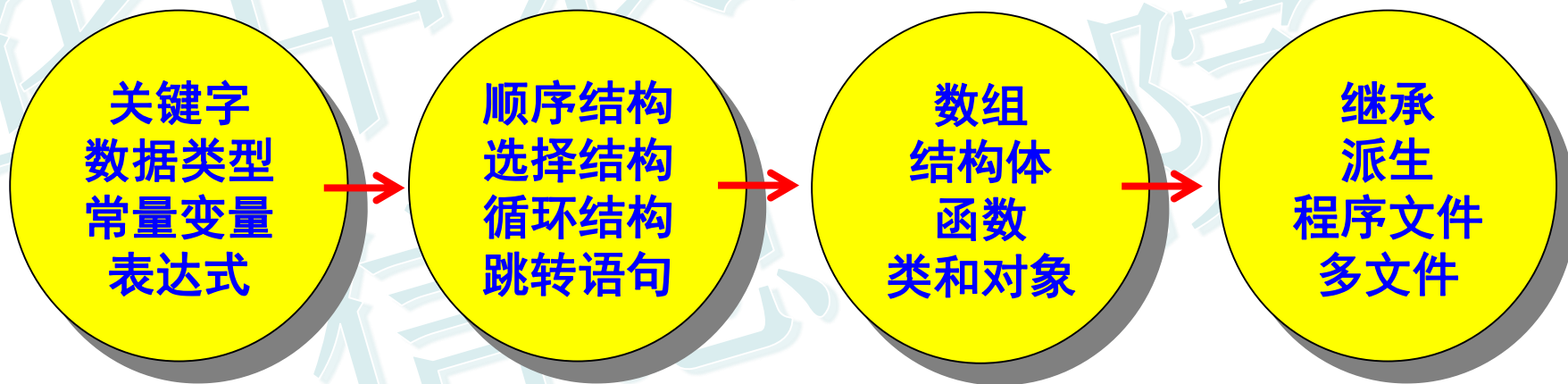


C/C++程序设计案例实战

——流水不腐户枢不蠹

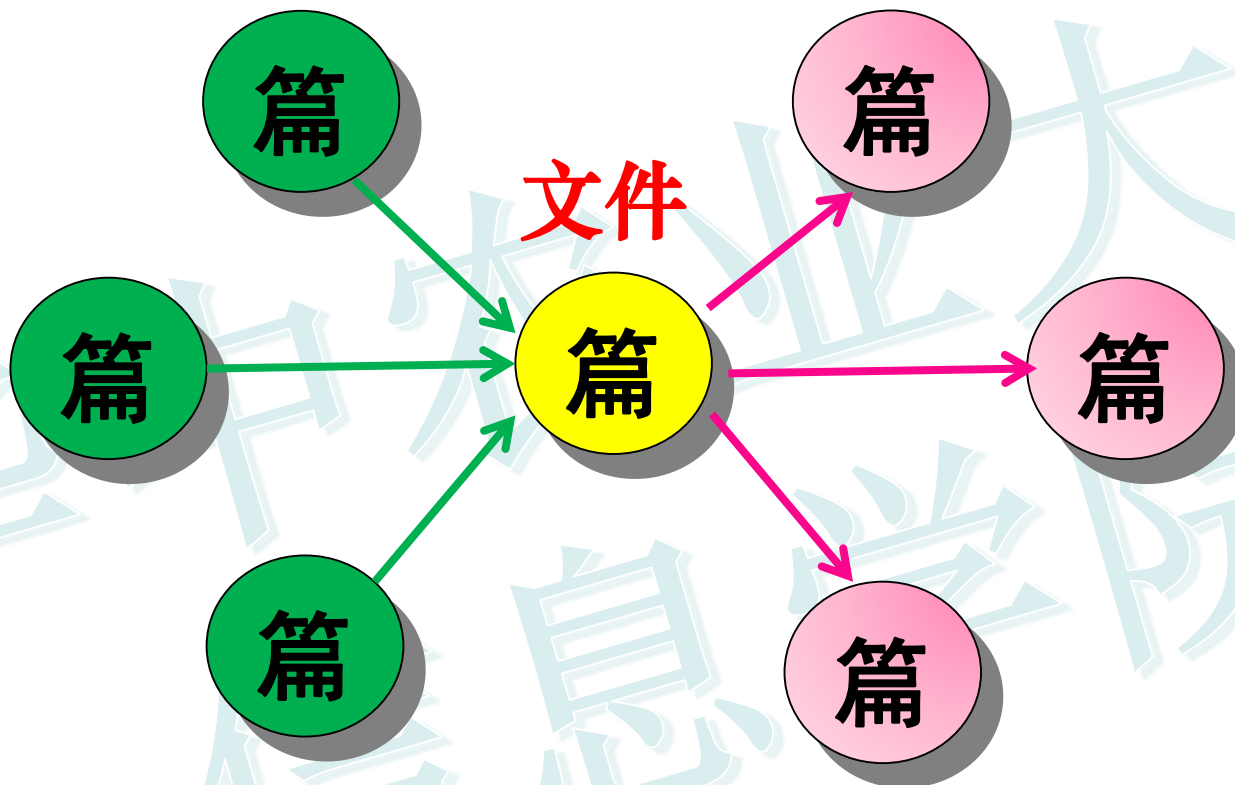
华中农业大学信息学院 章 英

问题引入



综合性设计性的程序项目

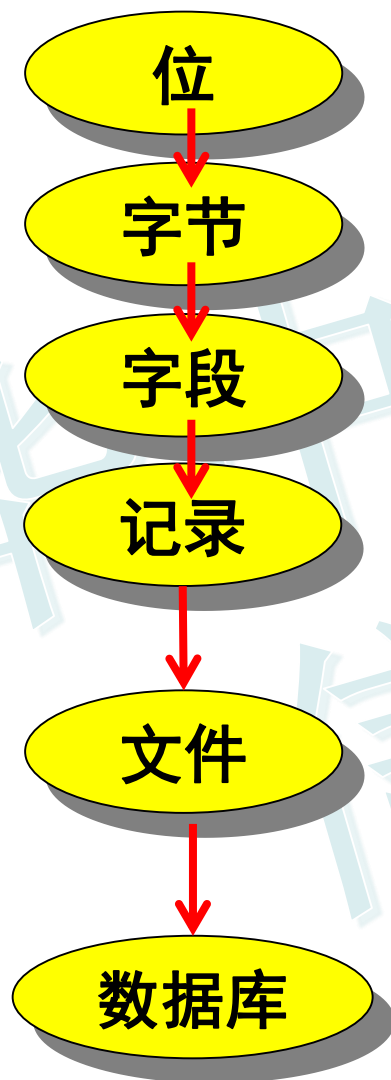
问题引入



数据输入

数据输出

计算机中的数据层次



Bit: 计算机按位模式操作和处理数据

Byte: 程序员以字节为单位申请数据

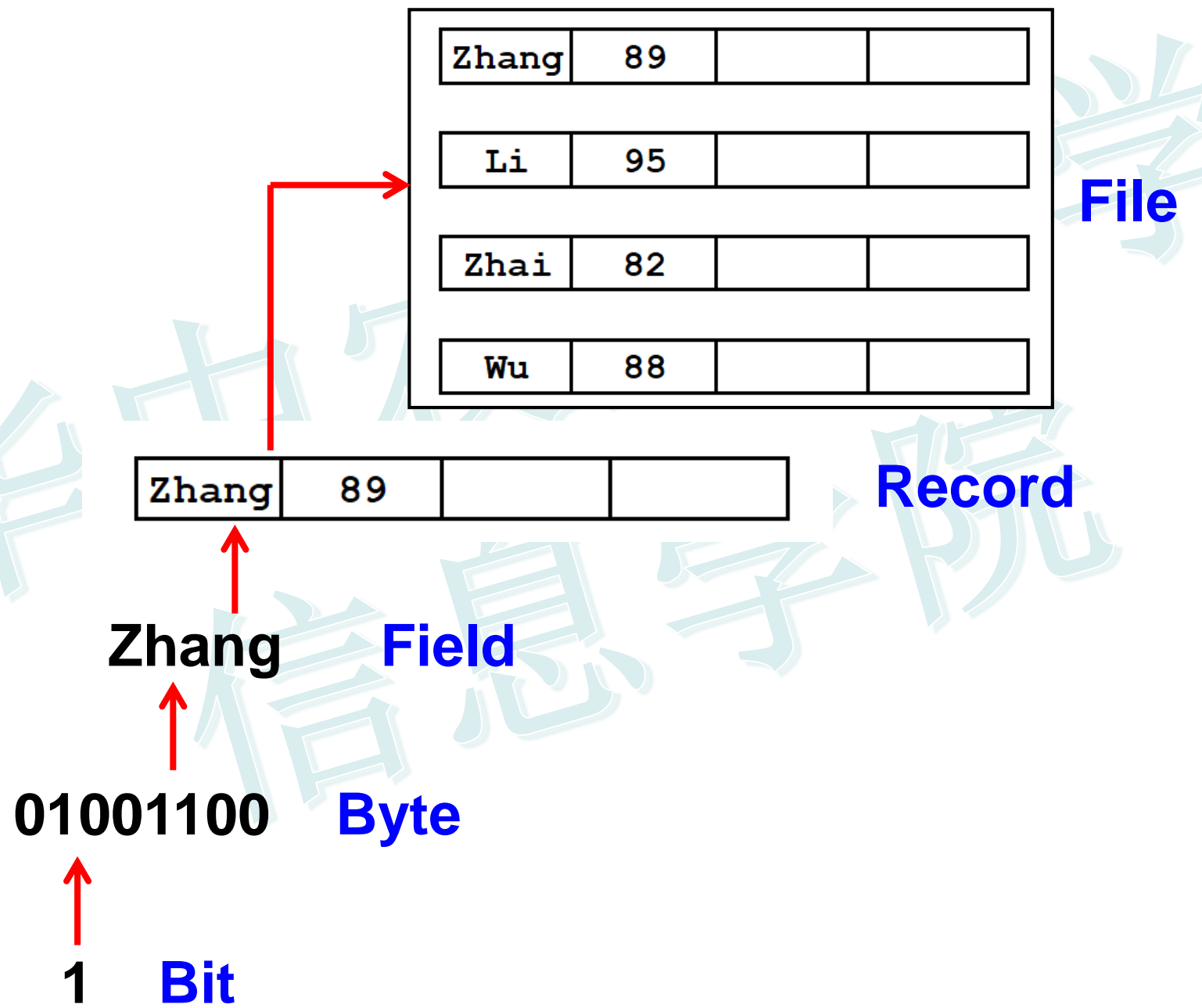
Field: 一组有意义的字符，比如name

Record: 一些有意义的字段的集合，比如班级类，某学生的信息

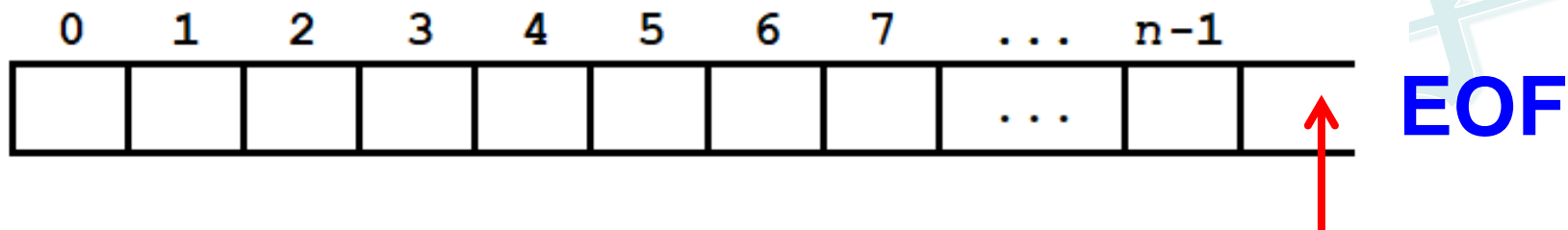
File: 相关记录的集合，比如班级学生成绩统计表

Database: 相关文件的集合，比如全校学生信息

计算机中的数据层次



文件：字节流



Ends with *end-of-file* marker

什么是流?

数据从一个对象流动到另一个对象，这种流动抽象为流，例如：`cin >> x`、`cout << x`;

水流

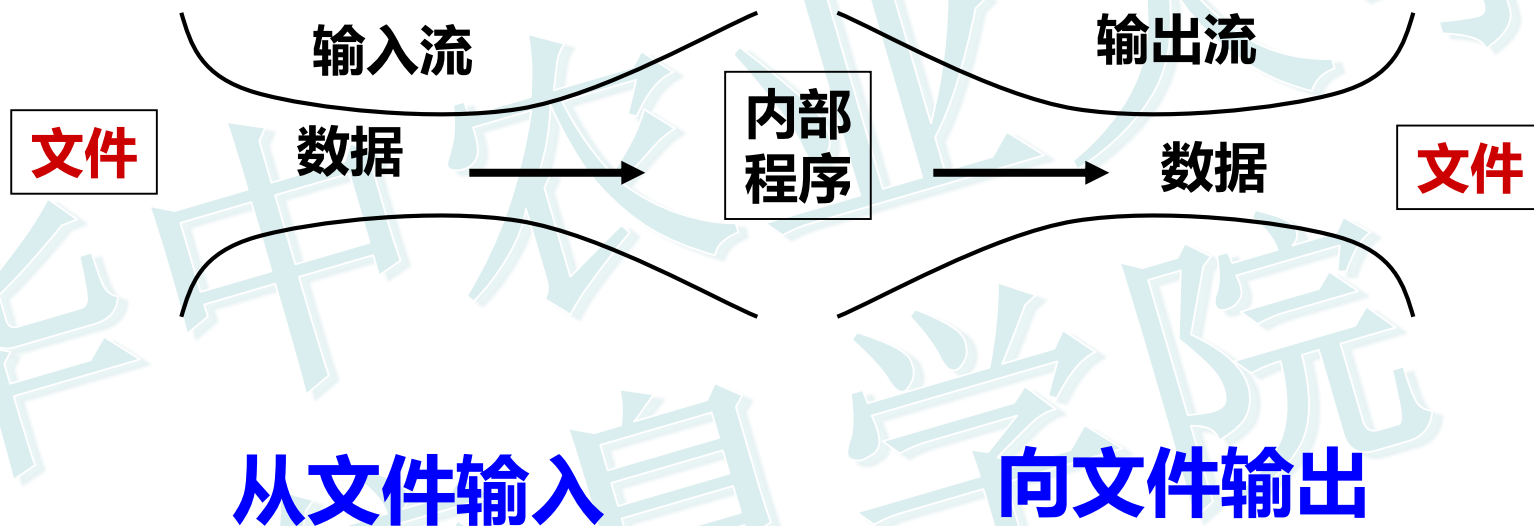


流水不腐，户枢不蠹，动也。

——吕氏春秋·尽数



什么是流?



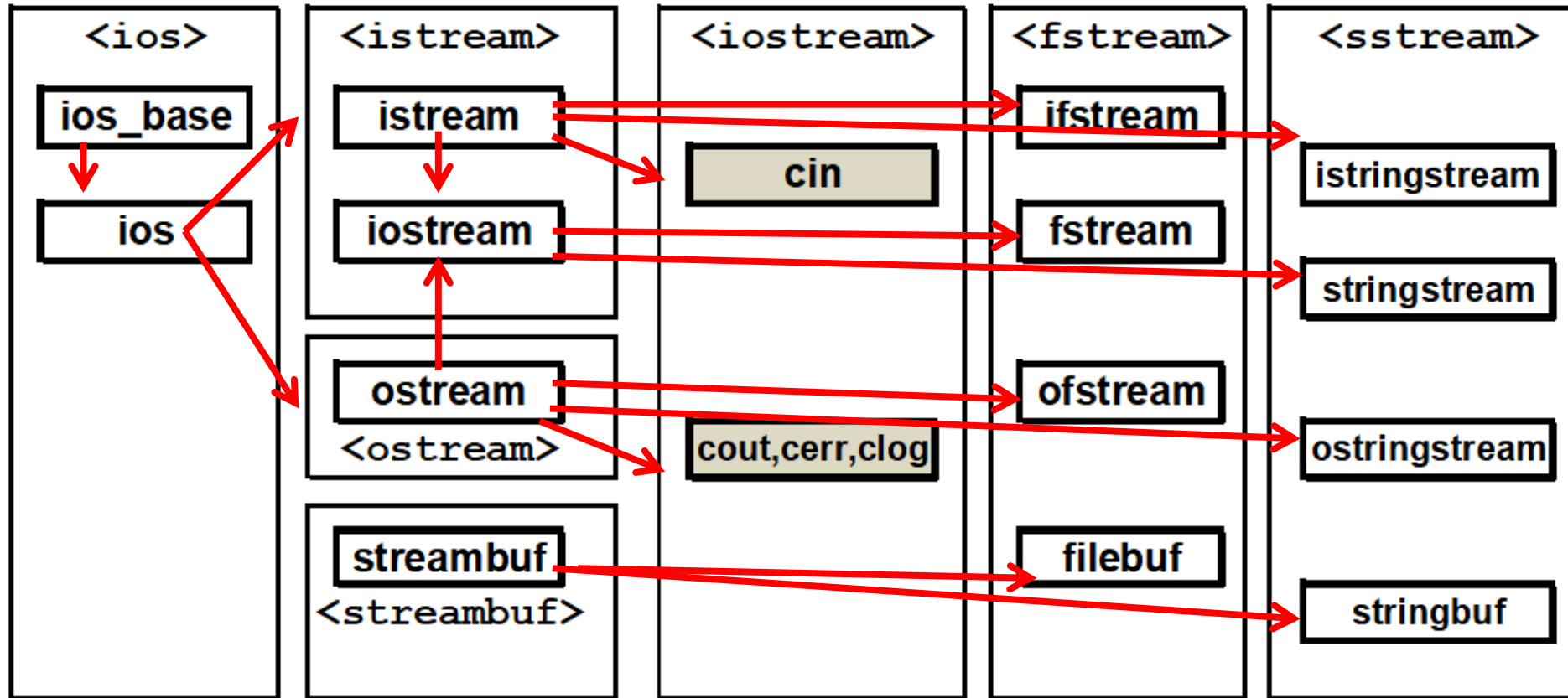
流的分类

标准I/O流：内存与标准输入输出设备之间信息的传递；

文件I/O流：内存与外部文件之间信息的传递；

字符串I/O流：内存变量与表示字符串流的字符数组之间信息的传递

I/O流类层次图



案例1——代码实现

- 使用I/O流以文本方式建立一个文件test1.txt，写入字符串“已成功写入文件！”，用其他字处理程序打开，看看是否正确写入。

```
#include <fstream>
using namespace std;
int main() {
    ofstream file("test1.txt");
    file.write("已成功写入文件!",16);
    //file<<"已成功写入文件! ";
    file.close();
    return 0;
}
```

“输出流”对象的构造过程

文本文件
和二进制
文件的区
别？

■ 构造输出流对象

➤ `ofstream file("filename");`

或者

➤ `ofstream file; // 建立一个文件流对象`

➤ `file.open("filename");`

■ 打开文件时可以指定模式

➤ `ofstream file("filename", ios_base::out | ios_base::binary);`

■ 使用插入(<<)运算符传送字节到一个输出流对象

■ 格式操纵符：setw、setiosflags、setprecision

■ 文件输出流的成员函数：`open、put、write、close`等

案例2——代码实现

- 使用I/O流以文本方式打开上一个案例建立的文件 test1.txt，读出其内容并显示出来。

```
1  #include <fstream>
2  #include <iostream>
3  using namespace std;
4  int main() {
5      char ch;
6      ifstream is("test1.txt");
7      while(is.get(ch))
8          cout<<ch;
9      is.close();
10     return 0;
11 }
12
```

已成功写入文件!

“输入流”对象的构造过程

■ 构造输入流对象

➤ **ifstream** myFile("filename");

或者

➤ **ifstream** myFile; // 建立一个文件流对象

➤ myFile.open("filename");

■ 打开文件时可以指定模式

➤ **ifstream** myFile("filename", **ios_base::in** | ios_base::binary);

■ 使用**提取(>>)**运算符传送字节到一个输入流对象

■ 成员函数：open、get、getline、read、close等

案例3——代码实现

- 使用I/O流以文本方式打开上一个案例建立的文件 test1.txt，在文件后面添加字符串“end of file!”，然后读出整个文件的内容显示出来。

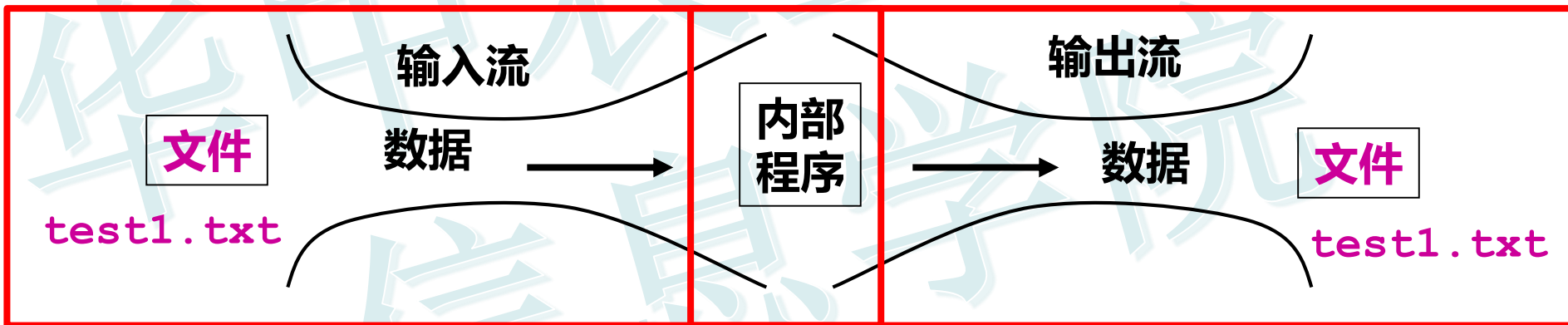
```
1  #include <fstream>
2  #include <iostream>
3  using namespace std;
4  int main() {
5      ofstream file1("test1.txt", ios::app);
6      file1<<"end of file!";
7      file1.close();
8      char ch;
9      ifstream file2("test1.txt");
10     while(file2.get(ch))
11         cout<<ch;
12     file2.close();
13     return 0;
14 }
15
```

已成功写入文件! end of file!

案例3——代码实现

ifstream 类 ofstream

file2 对象 file1



从文件输入

向文件输出

小结

- (1) 能够说出流的三种分类的名称
- (2) 能够画出I/O流类库的简单继承关系图
- (3) 能够编写文件输入流和文件输出流代码实现文件读写操作

延伸

编写程序实现：打开指定的一个文本文件，在每一行前加上行号后，将其输出到另一个文本文件中。

对比：标准I/O流、字符串I/O流