

实验 4 模板及 I/O 流管理

【实验目的】

- (1) 掌握模板函数和模板类的定义和使用方法。
- (2) 掌握标准输入输出的使用及格式控制方法。
- (3) 掌握磁盘文件（如二进制文件、文本文件）的输入输出的方法。

【实验内容】

1. 栈是一种重要的数据结构，它是一种只允许在表的一端进行插入或删除操作的线性表。表中允许进行插入、删除操作的一端称为栈顶。表的另一端称为栈底。栈顶的当前位置是动态的，对栈顶当前位置的标记称为栈顶指针。当栈中没有数据元素时，称之为空栈。栈的插入操作通常称为进栈或入栈，栈的删除操作通常称为退栈或出栈。下面是一个整型栈类的定义：

```
const int MaxSize = 100; // 栈中能保存的最多元素个数

class IStack {
public:
    IStack();           // 栈的构造函数
    void Push(int &n);   // 往栈顶增加元素
    void Pop();          // 从非空栈的栈顶删除一个元素
    int GetTop();        // 返回非空栈的栈顶元素
    bool Empty();        // 判断栈是否为空
    int Size();          // 返回栈中元素的个数
    void ClearStack();   // 将栈清空
    ~IStack();           // 栈的析构函数
private:
    int elem[MaxSize];   // 保存栈中各元素的数组
    int Top;             // 保存栈顶的当前位置
}
```

试编写栈的类模板（包括其成员函数的实现），以便为任何类型的对象提供栈结构的数据操作。然后在 `main()` 中实现整数栈和字符栈的各种操作。

2. 编写一个程序，实现以下功能：

（1）输入一系列的学生成绩（包括学号、姓名、成绩等数据）存放在文件 `stud.dat` 中。

（2）从 `stud.dat` 文件中读出这些数据并显示出来。

（3）在 `stud.dat` 文件中按姓名进行查询，如输入“李”，则将所有姓李的学生的数据都显示出来。

所编写的程序运行结果示例如下：

选择(1:输入数据 2:输出数据 3:按姓名查找数据 其他退出):1

输入数据

学生人数:5

第 1 个学生(学号 姓名 成绩):1001 张三 89

第 2 个学生(学号 姓名 成绩):1002 李四 78

第 3 个学生(学号 姓名 成绩):1003 王五 92

第 4 个学生(学号 姓名 成绩):1004 李沅芷 88

第 5 个学生(学号 姓名 成绩):1005 赵六 56

选择(1:输入数据 2:输出数据 3:按姓名查找数据 其他退出):2

输出数据

学号 姓名 成绩

1001 张三 89

1002 李四 78

1003 王五 92

1004 李沅芷 88

1005 赵六 56

选择(1:输入数据 2:输出数据 3:按姓名查找数据 其他退出):3

输入姓名:李

输出匹配的结果:

学号 姓名 成绩

1002 李四 78

1004 李沅芷 88

选择(1:输入数据 2:输出数据 3:按姓名查找数据 其他退出):3

输入姓名:李四

输出匹配的结果:

学号 姓名 成绩

1002 李四 78

选择(1:输入数据 2:输出数据 3:按姓名查找数据 其他退出):0

【实验指导】

1. 定义类模板时要注意模板形参 `T` 的说明方法；类对象定义的方法 `IStack<int>` 和 `IStack<char>`。请思考什么情况下 `IStack<T>` 需要定义特化模板？
2. 题（2）在进行二进制文件读写时，注意文件的打开方式应设为 `ios::binary`。