

# 实验 1 顺序表操作

## 一 实验目的

1. 掌握使用 VC++ 上机调试线性表的基本方法；
2. 掌握线性表顺序存储结构上的基本操作：插入、删除、查找以及合并有序表等运算在顺序存储结构上的实现。
3. 提交 OJ 系统进行验证（语言选择 C++）。

## 二 实验要求

1. 认真阅读并理解教材上相关操作函数。
2. 正确编写本程序并能通过 OJ 系统验证。
3. 必须完成：
  - 1) 认真阅读定义顺序表结构
  - 2) 认真阅读初始化顺序表
  - 3) 完成定位操作
  - 4) 完成插入操作
  - 5) 完成删除操作
  - 6) 完成翻转表
  - 7) 阅读主函数
4. 选择完成：
  - 1) 合并有序表

## 三 程序框架参考如下

### 第 1 题 顺序表的插入和删除（必做题）

**题目描述：**顺序表是数据结构中一种最基本的数据结构，它是用连续存储空间结构实现的线性表。主要包括顺序表操作，阅读顺序表数据结构定义、阅读顺序表的初始化、阅读顺序表的输出、完成插入操作、完成删除操作。

**输入：**构造一个顺序表，输入数据只有 3 行，第 1 行是一个整数  $n$ ，表示顺序表元素个数；第 2 行是  $n$  个整数，用这  $n$  个整数构造顺序表；第 3 行表示删除的数据元素的位置。

**输出：**依次输出顺序表中数据元素，每个元素以换行结束；然后输出被删除的数据元素数值；然后输出被删除后的顺序表的所有元素；最后输出被删除后顺序表的长度。

**输入样例：**

3

10 20 30 2
<b>输出样例</b> 10 20 30 20 10 30 2
<b>程序框架</b> 参见文件“实验 1-1 代码.cpp”，将出现“补充代码”的部分补充完整。

## 第 2 题 顺序表的查找和翻转（必做题）

<b>题目描述：</b> 顺序表是数据结构中一种最基本的数据结构，它是用连续存储空间结构实现的线性表。主要包括顺序表操作，阅读顺序表数据结构定义、阅读顺序表的初始化、阅读顺序表的输出、完成查找操作、完成翻转操作。
<b>输入：</b> 构造一个顺序表，输入数据只有 4 行，第 1 行是一个整数 $n$ ，表示顺序表元素个数；第 2 行是 $n$ 个整数，用这 $n$ 个整数构造顺序表；第 3 行和第 4 行输入两个待查找的数据元素的值。
<b>输出：</b> 依次输出顺序表中数据元素，每个元素以换行结束。然后输出 2 个被查找元素的位置（未找到返回 0）。最后输出被翻转后的顺序表的所有元素。
<b>输入样例：</b> //input 4 11 22 33 44 22 55
<b>输出样例</b> 11 22 33 44 2 0 44 33 22 11
<b>程序框架</b> 参见文件“实验 1-2 代码.cpp”，将出现“补充代码”的部分补充完整。

### 第 3 题 顺序表的合并（选做题）

<b>题目描述：</b> 顺序表是数据结构中一种最基本的数据结构，它是用连续存储空间结构实现的线性表。主要包括顺序表操作，阅读顺序表数据结构定义、阅读顺序表的初始化、阅读顺序表的输出、完成合并操作。
<b>输入：</b> 构造一个顺序表，输入数据只有 4 行，第 1 行是一个整数 $n$ ，表示顺序表元素个数；第 2 行是 $n$ 个非递减整数，用这 $n$ 个整数构造顺序表 $la$ ；第 3 行是一个整数 $m$ ，表示顺序表元素个数；第 4 行是 $m$ 个非递减整数，用这 $m$ 个整数构造顺序表 $lb$ 。
<b>输出：</b> 依次输出 $la$ 和 $lb$ 合并后的顺序表 $lc$ 中所有元素。
<b>输入样例：</b> 3 5 7 7 2 3 8
<b>输出样例</b> 3 5 7 7 8
<b>程序框架</b> 参见文件“实验 1-3 代码.cpp”，将出现“补充代码”的部分补充完整。