1. **线性表可选题目**
   1. 考试报名管理
      1. 项目简介
      2. 设计思路
      3. 数据结构
      4. 程序清单
      5. 运行结果

# 第一章 线性表

线性表是数据结构中最简单、最常用的一种线性结构。本次实验通过**模拟实际项目**来学习线性表的顺序和链式存储结构，通过使用有关线性表的操作实现考试报名管理。

## 1.1 考试报名管理

### 1.1.1 项目简介

考试报名给各高校报名工作带来了新的挑战，给教务管理部门增加了很大的工作量，报名数据手工录入既费时又会不可避免地出现错误。本项目是对考试报名管理的简单模拟，请用**菜单选择方式**完成下列功能：输入考生信息；输出考生信息；查询考生信息；添加考生信息；修改考生信息；删除考生信息。

### 1.1.2 设计思路

本项目的实质是完成对考生信息的建立、查找、插入、修改、删除等功能，首先定义项目的数据结构，然后将每个功能写成一个**函数**来完成对数据的操作，最后完成**主函数**以验证各个函数功能并得出运行结果。

### 1.1.3 数据结构

本项目的数据是考生信息，每条考生信息由准考证号、姓名、性别、年龄、报考专业等信息组成，这组考生信息具有相同特性，属于同一数据对象，相邻数据元素之间存在序偶关系。由此可以看出，这些数据也具有线性表中数据元素的性质，所以该系统的数据可以采用线性表来存储。

线性表的**顺序存储结构**的特点是逻辑关系相邻的两个元素在物理位置上也相邻，因此可以随机存储表中任一元素，它的存储位置可用一个简单、直观的公式来表示。然而，从另一个方面来看，这个特点也铸成了这种存储结构的弱点：在做插入或删除操作时，需要移动大量元素。为克服这一缺点，我们引入另一种存储形式――**链式存储**。链式存储是线性表的另一种表示方法，由于它不要求逻辑上相邻的元素在物理位置上也相邻，因此它没有顺序存储结构的弱点，但同时也失去了顺序表可随机存取的特点。

链式存储的优点是插入或删除元素时很方便，使用灵活。缺点是存储密度小，存储空间利用率低。事实上，链表插入、删除运算的快捷是以空间代价来换取时间。

顺序表适宜于做查找这样的静态操作；链表宜于做插入、删除这样的动态操作。若线性表的长度变化不大，且其主要操作是查找，则采用顺序表；若线性表的长度变化较大，且其主要操作是插入、删除操作，则采用链表。

本项目如果采用链式存储结构。可以用结构体类型定义每个考生信息，该单链表中的每个结点的结构可描述为（仅供参考）：

typedef struct examinee

{ char examno[10]; //准考证号

char name[10]; //姓名

char sex; //性别

float age; //年龄

char examtype[10]; //报考专业

} ElemType;

**注意事项：**

**1.实验报告见群文件的模板，注意仔细阅读其中的格式要求。**

**2.实验报告1上传到群文件，截止到2020.02.26夜22：00。**

**3.我们此门课程的目的是让同学们从底层了解数据的组织和使用，所以禁止使用任何数据库存储数据，可以（建议）在外部文本文件（如.txt等文件）中存储数据。**

**4.实验报告命名格式(word或者压缩文件)：实验1-学号-姓名 例如：**

**实验1-2005005888-张三.doc**

**实验1-2005005888-张三.zip**