一.数据结构与算法概述

本课程由数据结构+算法构成，数据结构的要求相对简单，而算法则是在数据结构的基础上进一步的学习。有一些问题可能只需要数据结构的知识就能解决，但更多的问题还需要在掌握数据结构的基础上掌握算法才能解决。

二.数据结构

1.线性结构

是最常用的数据结构，特点是数据与元素之间存在**一对一的线性关系**（如a[0]=30)。线性结构有两种不同的存储结构，分为顺序存储结构和链式存储结构。顺序存储的线性表成为顺序表，顺序表的存储元素是连续的（指**地址是连续**的）。而链式存储的线性表成为链表，链表的存储元素的地址不一定连续，元素节点**存放数据元素和相邻元素的地址信息**，不连续的好处在于可以利用**碎片化的地址存储元素。**

常见的线性结构有：**数组，队列，链表，栈。**

2.非线性结构

不一定是一对一的关系。

常见的非线性结构：**二维数组，多维数组，广义表，树结构，图结构。**

三.稀疏数组和队列

1.基本介绍

当一个数组中大部分元素为0或者为同一个值的数组时，可以使用稀疏数组来保存数据。

稀疏数组的处理方法：

（1）记录数组一共有几行几列，几个不为0的值；

（2）把具有不同值的行，列，值记录在小规模数组中，这个小规模数组就是稀疏数组。

一个典型的应用场景就是记录五子棋盘，原始的二维数组会存在很多没有意义的数据而使用稀疏数组会起到压缩的效果。

2.将二维数组转化为稀疏数组

思路：

1.从二维数组中得到非0元素的个数count，从而创建稀疏数组int[count+1][3]；

2.再次遍历二维数组，把每一个非0元素的行列值保存到稀疏数组中；

3.将稀疏数组转化为二维数组

思路：

1.从稀疏数组的第一行中获得行列信息，从而初始化二维数组int[row][column]；

2.把稀疏数组后面几行的信息转化为二维数组的数据