

91#294

选择排序

通过 $n-i$ 次关键字之间的比较,从 $n-i+1$ 个记录中选择关键字最小的记录,并和第 $i(1 \leq i \leq n)$ 个记录交换之

尽管与冒泡排序同为 $O(n^2)$,但简单选择排序的性能要略优于冒泡排序

算法内容是选择排序，具体实现方式采用链表。

92#295

冒泡排序

基本思想是:两两比较相邻记录的关键字,如果反序则交换

冒泡排序时间复杂度最好的情况为 $O(n)$,最坏的情况是 $O(n^2)$

改进思路1: 设置标志位, 明显如果有一趟没有发生交换 ($flag = false$), 说明排序已经完成

改进思路2: 记录一轮下来标记的最后位置, 下次从头部遍历到这个位置就Ok

93#296

交换排序实质上就是选择排序，是91题在具体问题中的应用（砾石）

94#297

正常从小到大进行排序即可，就是数组分割的有序版。

95#298

快速排序

通过一趟排序将要排序的数据分割成独立的两部分，其中一部分的所有数据都比另外一部分的所有数据都要小，然后再按此方法对这两部分数据分别进行快速排序，整个排序过程可以递归进行，以此达到整个数据变成有序序列。时间复杂度为 $O(n \log n)$

排序完之后查找一下即可

96#299

计数排序

算法的步骤如下：

- (1) 找出待排序的数组中最大和最小的元素
- (2) 统计数组中每个值为*i*的元素出现的次数，存入数组C的第*i*项
- (3) 对所有的计数累加（从C中的第一个元素开始，每一项和前一项相加）
- (4) 反向填充目标数组：将每个元素*i*放在新数组的第C(*i*)项，每放一个元素就将C(*i*)减去1

97#301

数组的循环座椅，直接暴力模拟即可。

98#302

遍历时统计一遍出现次数最多的元素即可。

99#303

这题困扰了我一会儿.....

首先讲一下我的算法思路，先将整体数组从小到大排序，那么从1- $n/2$ 的全分给小的集合， $n/2+1$ - n 的全分给大的集合就可以保证无论*n*为奇数偶数的情况下，都有 $|n1-n2|$ 最小且 $|S1-S2|$ 最大的性质。

后来我发现这样做有Bug，虽然题目没给

100#304

WPL我知道，二叉树我也知道.....但我怎么觉得这题的样例我看不太懂.....

如果有更清楚的题意的话我再写= =