**第二课大纲**

# 排序

## 插入排序（简单插入排序）

插入排序（Insertion-Sort）的算法描述是一种简单直观的排序算法。它的工作原理是通过构建有序序列，对于未排序数据，在已排序序列中从后向前扫描，找到相应位置并插入。

S1 从第一个元素开始，该元素可以认为已经被排序；

S2 取出下一个元素，在已经排序的元素序列中从后向前扫描；

S3 如果该元素（已排序）大于新元素，将该元素移到下一位置；

S4 重复步骤3，直到找到已排序的元素小于或者等于新元素的位置；

S5 将新元素插入到该位置后；

S6 重复步骤2~5。

## 冒泡排序

冒泡排序是一种简单的排序算法。它重复地走访过要排序的数列，一次比较两个元素，如果它们的顺序错误就把它们交换过来。走访数列的工作是重复地进行直到没有再需要交换，也就是说该数列已经排序完成。这个算法的名字由来是因为越小的元素会经由交换慢慢“浮”到数列的顶端。

S1 比较相邻的元素。如果第一个比第二个大，就交换它们两个；

S2 对每一对相邻元素作同样的工作，从开始第一对到结尾的最后一对，这样在最后的元素应该会是最大的数；

S3 针对所有的元素重复以上的步骤，除了最后一个；

S4 重复步骤1~3，直到排序完成。

## 快速排序

快速排序的基本思想：通过一趟排序将待排记录分隔成独立的两部分，其中一部分记录的关键字均比另一部分的关键字小，则可分别对这两部分记录继续进行排序，以达到整个序列有序。

S1 从数列中挑出一个元素，称为 “基准”（pivot）；

S2 重新排序数列，所有元素比基准值小的摆放在基准前面，所有元素比基准值大的摆在基准的后面（相同的数可以到任一边）。在这个分区退出之后，该基准就处于数列的中间位置。这个称为分区（partition）操作；

S3 递归地（recursive）把小于基准值元素的子数列和大于基准值元素的子数列排序。

## 经典问题：数组第K小

# 二分

二分查找也称折半查找（Binary Search），它是一种效率较高的查找方法。但是，折半查找要求线性表必须采用顺序存储结构，而且表中元素按关键字有序排列。

首先，假设表中元素是按升序排列，将表中间位置记录的关键字与查找关键字比较，如果两者相等，则查找成功；否则利用中间位置记录将表分成前、后两个子表，如果中间位置记录的关键字大于查找关键字，则进一步查找前一子表，否则进一步查找后一子表。重复以上过程，直到找到满足条件的记录，使查找成功，或直到子表不存在为止，此时查找不成功。

## 二分的特点

给定谓词后，满足某个该谓词的答案区间连续，且不满足该谓词的答案区间也连续。

## 二分的实现

如何确定谓词？

如何表达区间？

如何确定初始区间？

如何确定中间值？

如何判断二分结束？

## 经典问题：最大值最小化