

01-D-4 取非极端元素 & 冒泡排序

#数据结构邓神

取非极端元素 (既不是最大, 也不是最小)

Q: 给定整数子集 S , $|S| = n \geq 3$

找出元素 a 属于 S , 且 $a \neq \max(S)$ 且 $a \neq \min(S)$

算法: 从 S 中任意取出三个元素

// 如果 S 用数组的形式给出 不妨取出前三个

// 由于 S 是集合, 所以这个三个元素必然互异

确定并且排除其中的最大最小值

// 不妨设 $x = \max\{x, y, z\}$ $y = \min\{x, y, z\}$

输出 z 即可

无论输入规模 n 多大, 上述算法所需要的执行时间不变

$T(n) = \text{常数} = O(1) = \Omega(1) = \Theta(1)$

冒泡排序

Q: 给定 n 个整数, 将他们按照非降序排序

观察: 有序/无序序列中, 任意/总有一对相邻元素顺序/逆序

扫描交换: 依次比较每一对相邻元素, 如果逆序, 交换之。

如果整趟扫描都没有出现交换, 则排序完成, 否则继续从头扫描

// 代码

```
void bubbleSort(int A[], int n){
    for(bool sorted=false; (sorted = !sorted); n--){ // sorted 表示 序列是否有序
        for (int i = 1; i < n; ++i) {
            if (A[i] < A[i-1]){ // 逆序交换之
                swap(A[i], A[i-1]);
                sorted = false; // 清除有序标识
            }
        }
    }
}
```

```
}  
  }  
}
```