

1.(15 分) 设 $A[1...n]$ 和 $B[1...n]$ 是两个已按照升序排列的数组(A 、 B 数组内部无相同元素)。假设我们的问题中, n 为 2 的整数次幂。合并集合 A 和 B , 并按升序排列得到集合 C 。请回答如下问题:

- (1) 假设 A 的中项为 a , B 的中项为 b (数组中下标为 $n/2$ 的元素), 那么 C 的中项和 a 、 b 之间有什么关系(提示: 分 $a=b$, $a<b$ 和 $a>b$ 讨论)。
- (2) 设计一个分治算法, 输入集合 A 和 B , 找到它们并集的中项, 写出其主要思想;
- (3) 分析算法的时间复杂度。

答案:

- (1) 在两个大小为 n 的数组 A, B 中取中位数, 分别为 a, b , 所用时间为 常数, 时间复杂度为 $\Theta(1)$;
- (2) 比较 a 和 b 的大小:
 - 若 $a=b$, return a , 即返回 a 为中位数;
 - 若 $a<b$, 则保留 A 中大于等于其中位数的部分, 和 B 的小于等于其中位数的部分, 分别组成新的数组, 重新调用自己;
 - 若 $a>b$, 则取保留 A 的小于等于其中位数部分, 和 B 中大于等于其中位数部分, 分别组成新的数组, 重新调用自己;按照上述算法执行, 直到找到最后每个数组只剩 1 个元素, 取小的元素作为中值;
- (3) 所需时间 $T(n)$ 满足: $T(n) = T(n/2) + \Theta(1)$, 时间复杂性为: $\Theta(\log(n))$

2、对于仅包含英文字母的数组 $A[1...n]$, 要求采用分治方法找出其中的“连续字母对”的个数。

说明: “连续字母对”即数组 $A[1...n]$ 中相邻的两个字母 $A[i]$ 与 $A[i+1]$ ($1 \leq i < n$) 存在前后连续的关系 (如: 字母 a 与 b 连续, a 与 c 不连续), 那么我们称 $A[i]$ 与 $A[i+1]$ 是一对连续字母对, 请求出数组 $A[1...n]$ 中连续字母对的个数。(总分 15)

- (1) 设计分治算法求出数列 $A[1...n]$ 中字母连续对的个数, 请说明设计思路 (3 分), 写出伪代码; (3 分)
- (2) 求出数组 $[a, e, f, g, b, i]$ 的连续字母对的个数; (4 分)
- (3) 简要说明采用算法在数组 $[a, e, f, g, b, i]$ 中进行查找的执行过程。(5 分)

参考答案:

- (1) 设计思路:
 - 整个问题可分解为三个子问题:
 - 第一个: 计算 $A[0] \sim A[n/2]$ 序列中的连续字母对;
 - 第二个: 计算 $A[n/2+1] \sim A[n]$ 的连续字母对,

第三个: 以上两个序列之间的连续字母对计算。

采用归并排序的思路, 首先递归求解计算前后两个字母段的连续字母对; 然后考虑第三个问题的求解, 我们仅仅需要对比 $A[n/2]$ (为左序列最后一个字母) 与 $A[n/2+1]$ (为右序列第一个字母开始), 如果 $A[n/2]$, $A[n/2+1]$ 为连续字母对, 则总连续字母对等于前面字母段与后面字母段的连续字母对的和加 1。

伪代码:

输入: n 个元素的字母数组 $A[1 \cdots n]$ 。

输出: 连续字母对的个数;

```
int sort (A[], left, right)
{
    if (left >= right)
        return 0;
    int mid = (left + right) / 2 ; // 实际上是取底函数
    int num_left = sort (A[], left, mid) ;
    int num_right = sort (A[], mid + 1, right);
    if (A[mid] 与 A[mid + 1] 为连续字母对)
        return num_left + num_right + 1;
    else
        return num_left + num_right;
}
```

(2)

2 对

(3) 算法执行流程如下:

对应输入数组 $[a, e, f, g, b, i]$, 采用以上算法之后会将问题划分成先求解 $[a, e, f]$, $[g, b, i]$, 然后再进行划分为 $[a, e]$ 与 $[f]$; $[g, b]$ 与 $[i]$;

判断 1: $[a, e]$ 与 $[f]$, $[a, e]$ 不是字母连续对, 考虑前面数组的最后一位 e 与后面数组的第一位 f , 可判断 e, f 是一个字母连续对, 所以 $num_left = 1$,

判断 2: $[g, b]$ 与 $[i]$, $[g, b]$ 不是字母连续对, $[g, b]$ 与 $[i]$ 也没有字母连续对, 因此后面的 $num_right = 0$;

判断 3: $[a, e, f]$ 的最后一位与 $[g, b, i]$ 的第一位是字母连续对, 所以, 总数为: $num_left + num_right + 1 = 2$;