

1. 设  $A=(a_1, \dots, a_m)$  和  $B=(b_1, \dots, b_n)$  均为顺序表,  $A'$  和  $B'$  分别为  $A$  和  $B$  中除去最大共同前缀后的子表, 若  $A'=B'=\text{空表}$ , 则  $A=B$ ; 若  $A'=\text{空表}$ , 而  $B' \neq \text{空表}$ , 或者两者都不为空表, 且  $A'$  的首元小于  $B'$  的首元, 则  $A < B$ ; 否则  $A > B$ 。试写一个比较  $A, B$  大小的算法。

2. 试写一算法, 实现顺序表的就地逆置, 即利用原表的存储空间将线性表  $(a_1, a_2, \dots, a_n)$  逆置为  $(a_n, a_{n-1}, \dots, a_1)$ 。

3. 试写一算法在带头结点的单链表结构上实现线性表操作  $\text{LENGTH}(L)$ 。

4. 已知指针  $ha$  和  $hb$  分别指向两个单链表的头结点, 并且已知两个链表的长度分别为  $m$  和  $n$ , 试写一算法将这两个链表连接在一起(即令其中一个表的首元结点连在另一个表的最后一个结点之后)。假设指针  $hc$  指向连接后的链表的首元结点, 并要求算法以尽可能短的时间完成连接运算。请分析你的算法的时间复杂度。

5. 已知线性表的元素以值递增有序排列, 并以单链表作存储结构, 试写一高效的算法, 删除表中所有值大于  $\text{mink}$  且小于  $\text{maxk}$  的元素(若表中存在这样的元素), 同时释放被删结点空间, 并分析你的算法的时间复杂度。

6. 试写一算法, 对单链表实现就地逆置。

7. 假设某个单向循环链表的长度大于 1, 且表中既无头结点也无头指针。已知  $s$  指向链表中某个结点的指针, 试编写算法在链表中删除指针  $s$  所指结点的前驱结点。