

Q1:

假设有一个带头结点的单链表  $L = (a_1, b_1, a_2, b_2, \dots, a_n, b_n)$ 。设计一个算法将其拆分成两个带头结点的单链表  $L1$  和  $L2$ ：

$$L1 = (a_1, a_2, \dots, a_n), L2 = (b_n, b_{n-1}, \dots, b_1)$$

要求  $L1$  使用  $L$  的头结点。

Q2: 某线性表最常用的操作是在尾元素之后插入一个元素和删除第一个元素，故采用（ ）存储方式最节省运算时间。

- A. 单链表
- B. 仅有头结点指针的循环单链表
- C. 双链表
- D. 仅有尾结点指针的循环单链表

Q3: 如果对含有  $n$  ( $n > 1$ ) 个元素的线性表的运算只有 4 种，即删除第一个元素、删除尾元素、在第一个元素前面插入新元素、在尾元素的后面插入新元素，则最好使用（ ）。

- A. 只有尾结点指针没有头结点的循环单链表
- B. 只有尾结点指针没有头结点的非循环双链表
- C. 只有首结点指针没有尾结点指针的循环双链表
- D. 既有头指针也有尾指针的循环单链表

Q4: 一个长度为  $L$  ( $L \geq 1$ ) 的升序序列  $S$ ，处在第  $L/2$  个位置的数称为  $S$  的中位数。

例如：若序列  $S_1=(11, 13, 15, 17, 19)$ ，则  $S_1$  的中位数是 15。

两个序列的中位数是含它们所有元素的升序序列的中位数。例如，若  $S_2=(2, 4, 6, 8, 20)$ ，则  $S_1$  和  $S_2$  的中位数是 11。

现有两个等长的升序序列  $A$  和  $B$ ，试设计一个在时间和空间两方面都尽可能高效的算法，找出两个序列  $A$  和  $B$  的中位数。要求：

- (1) 给出算法的基本设计思想。
- (2) 根据设计思想，采用 C、C++ 或 Java 语言描述算法，关键之处给出注释。
- (3) 说明你所设计算法的时间复杂度和空间复杂度。

DDL: 20/09/2024