

一休直播讲解的题目

这些题目都是关于顺序表（数组）的真题，大家听直播之前最好能做一下，只做暴力算法，不要求写出最优解，但是一定要写出解法来，然后最好能把思路 and 想法记录下来，当然如果还有时间可以想想还有没有可以优化和进步的地方。可以自己复制截图或打印下来。

7.15

2010 年

42. (13分) 设将 n ($n > 1$) 个整数存放于一维数组 R 中。试设计一个在时间和空间两方面都尽可能高效的算法。将 R 中保存的序列循环左移 p ($0 < p < n$) 个位置，即将 R 中的数据由 (X_1, X_2, \dots, X_n) 变换为 $(X_{p+1}, X_{p+2}, \dots, X_n, X_1, X_2, \dots, X_p)$ 。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。

2011 年

42. (15分) 一个长度为 L ($L \geq 1$) 的升序序列 S ，处在第 $\lfloor L/2 \rfloor$ 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。现在有两个等长升序序列 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。

2013 年

41. (13分) 已知一个整数序列 $A = (a_1, a_2, \dots, a_n)$ ，其中 $0 \leq a_i < n$ ($0 \leq i < n$)。若存在 $a_p = a_{p+1} = \dots = a_k = x$ 且 $m > n/2$ ($0 \leq p < n$, $1 \leq k \leq m$)，则称 x 为 A 的主元素。例如 $A = (0, 5, 5, 3, 5, 7, 5, 5)$ ，则 5 为主元素；又如 $A = (0, 5, 5, 3, 5, 1, 5, 7)$ ，则 A 中没有主元素。假设 A 中的 n 个元素保存在一个一维数组中，请设计一个尽可能高效的算法，找出 A 的主元素。若存在主元素，则输出该元素；否则输出 -1。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。

2016 年

43. 已知由 n ($n \geq 2$) 个正整数构成的集合 $A = \{a | 0 \leq k < n\}$, 将其划分为两个不相交的子集 A 和 A , 元素个数分别是 n 和 n , A 和 A 中元素之和分别为 S 和 S . 设计一个尽可能高效的划分算法, 满足 $|n-n|$ 最小且 $|S-S|$ 最大。要求:

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

2018 年

41. (13 分) 给定一个含 n ($n \geq 1$) 个整数的数组, 请设计一个在时间上尽可能高效的算法, 找出数组中未出现的最小正整数。例如, 数组 $\{-5, 3, 2, 3\}$ 中未出现的最小正整数是 1; 数组 $\{1, 2, 3\}$ 中未出现的最小正整数是 4。要求:

- (1) 给出算法的基本设计思想。
- (2) 根据设计思想, 采用 C 或 C++ 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。

2020 年

41. (13 分) 定义三元组 (a, b, c) (a, b, c 均为正数) 的距离 $D = |a - b| + |b - c| + |c - a|$ 。给定 3 个非空整数集合 S_1, S_2 和 S_3 , 按升序分别存储在 3 个数组中。请设计一个尽可能高效的算法, 计算并输出所有可能的三元组 (a, b, c) ($a \in S_1, b \in S_2, c \in S_3$) 中的最小距离。例如 $S_1 = \{-1, 0, 9\}, S_2 = \{-25, -10, 10, 11\}, S_3 = \{2, 9, 17, 30, 41\}$, 则最小距离为 2, 相应的三元组为 $(9, 10, 9)$ 。要求:

- (1) 给出算法的基本设计思想。
- (2) 根据设计思想, 采用 C 或 C++ 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。

7.18

2009 年

42. (15 分) 已知一个带有表头结点的单链表, 结点结构为

data	link
------	------

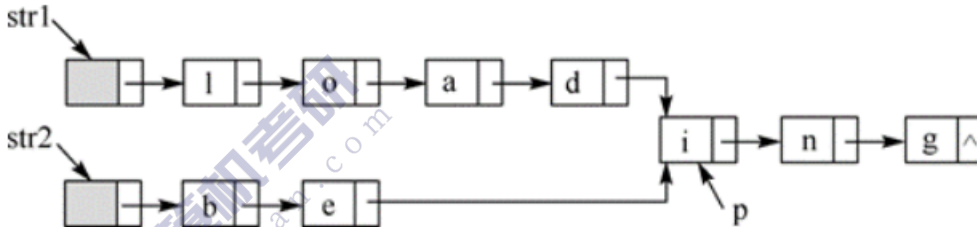
假设该链表只给出了头指针 list。在不改变链表的前提下, 请设计一个尽可能高效的算法, 查找链表中倒数第 k 个位置上的结点 (k 为正整数)。若查找成功, 算法输出该结点的 data 域的值, 并返回 1; 否则, 只返回 0。要求:

- 1) 描述算法的基本设计思想。
- 2) 描述算法的详细实现步骤。

3) 根据设计思想和实现步骤, 采用程序设计语言描述算法 (使用 C、C++ 或 Java 语言实现), 关键之处请给出简要注释。

2012 年

42. 假定采用带头结点的单链表保存单词, 当两个单词有相同的后缀时, 则可共享相同的后缀存储空间, 例如, "loading" 和 "being" 的存储映像如下图所示。



设 $str1$ 和 $str2$ 分别指向两个单词所在单链表的头结点, 链表结点结构为

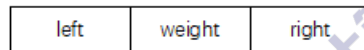
data	next
------	------

, 请设计一个时间上尽可能高效的算法, 找出由 $str1$ 和 $str2$ 所指向两个链表共同后缀的起始位置 (如图中字符 i 所在结点的位置 p)。要求:

- 1) 给出算法的基本设计思想。
- 2) 根据设计思想, 采用 C 或 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。

2014 年

41. (13 分) 二叉树的带权路径长度 (WPL) 是二叉树中所有叶结点的带权路径长度之和。给定一棵二叉树 T , 采用二叉链表存储, 结点结构如下:



其中叶结点的 $weight$ 域保存该结点的非负权值。设 $root$ 为指向 T 的根结点的指针, 请设计求 T 的 WPL 的算法, 要求:

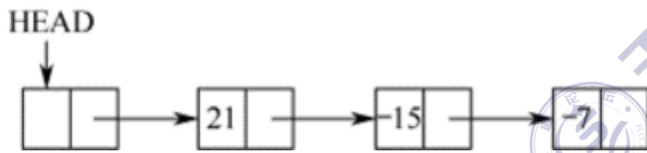
- 1) 给出算法的基本设计思想。
- 2) 使用 C 或 C++ 语言, 给出二叉树结点的数据类型定义。
- 3) 根据设计思想, 采用 C 或 C++ 语言描述算法, 关键之处给出注释。

2015 年

41. (15 分) 用单链表保存 m 个整数, 结点的结构为 $[data][link]$, 且 $|data| \leq n$ (n 为正整数)。现要求设计一个时间复杂度尽可能高效的算法, 对于链表中 $data$ 的绝对值相等的结点, 仅保留第一次出现的结点而删除其余绝对值相等的结点。例如, 若给定的单链表 HEAD 如下:



则删除结点后的 HEAD 为

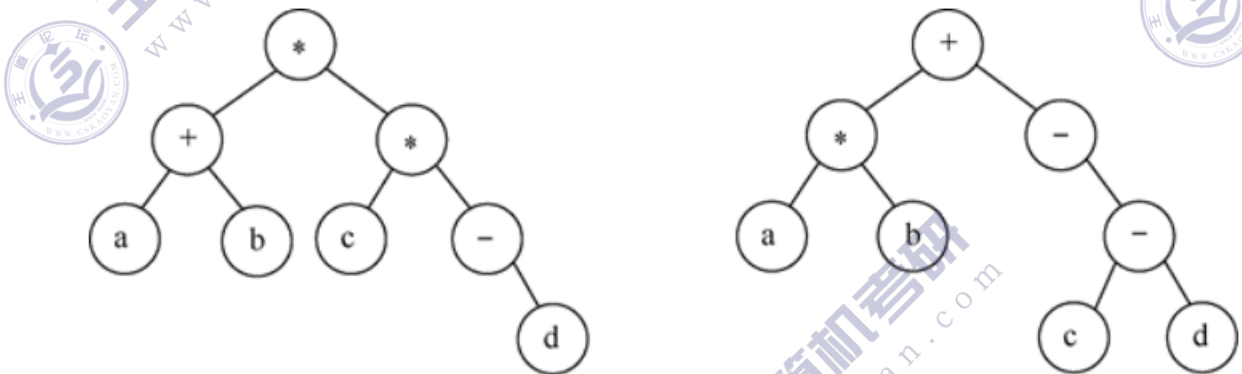


要求:

- 1) 给出算法的基本设计思想。
- 2) 使用 C 或 C++ 语言, 给出单链表结点的数据类型定义。
- 3) 根据设计思想, 采用 C 或 C++ 语言描述算法, 关键之处给出注释。
- 4) 说明你所设计算法的时间复杂度和空间复杂度。

2017 年

41. (15 分) 请设计一个算法, 将给定的表达式树 (二叉树) 转换为等价的中缀表达式 (通过括号反映操作符的计算次序) 并输出。例如, 当下列两棵表达式树作为算法的输入时, 输出的等价中缀表达式分别为 $(a+b)*(c*(-d))$ 和 $(a*b)+(-(c-d))$ 。



二叉树结点定义如下:

```
typedef struct node{
    char data[10]; //存储操作数或操作符
    struct node *left, *right;
}BTree;
```

要求:

- (1) 给出算法的基本设计思想。
- (2) 根据设计思想, 采用 C 或 C++ 语言描述算法, 关键之处给出注释。

2019 年

41. (13 分) 设线性表 $L = (a_1, a_2, a_3, \dots, a_{n-2}, a_{n-1}, a_n)$ 采用带头结点的单链表保存, 链表中的结点定义如下:

```
typedef struct node {
    int data;
```

```
struct node* next;
```

```
} NODE;
```

请设计一个空间复杂度为 $O(1)$ 且时间上尽可能高效的算法，重新排列 L 中的各结点，得

到线性表 $L' = (a_1, a_n, a_2, a_{n-1}, a_3, a_{n-2}, \dots)$ 。要求：

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

2021 年

41. (15 分) 已知无向连通图 G 由顶点集 V 和边集 E 组成， $|E| > 0$ ，当 G 中度为奇数的顶点个数为不大于 2 的偶数时， G 存在包含所有边且长度为 $|E|$ 的路径（称为 EL 路径）。设图 G 采用邻接矩阵存储，类型定义如下：

```
typedef struct { //图的定义
    int numVertices, numEdges; //图中实际的顶点数和边数
    char VerticesList[MAXV]; //顶点表。MAXV 为已定义常量
    int Edge[MAXV][MAXV]; //邻接矩阵
} MGraph;
```

请设计算法 `int IsExistEL(MGraph G)`，判断 G 是否存在 EL 路径，若存在，则返回 1，否则返回 0。要求：

- 1) 给出算法的基本设计思想。
- 2) 根据设计思想，采用 C 或 C++ 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。