

2026 考研数据结构代码题 基础 63 题做题本

来源：b 站：我头发还多还能学

微信：Rain—Splash 或 petrichoryin

强化 75 题、408 最终预测 15 题、上机运行脚本以及所有题目对应的
视频讲解需要付费，有需要的同学添加微信即可，希望对大家有帮助。
打印资料可以直接扫右侧二维码，价格无敌便宜，纸张质量也很棒



基础篇目录

顺序表

- 1、在顺序表 L 的第 k 个位置插入元素 x 【增】
- 2、删除顺序表 L 的第 k 个元素并返回其值 【删一】
- 3、将顺序表中的元素逆置 【改：经典逆置】
- 4、查找顺序表中第一个值为 x 的元素的位置 【查】
- 5、顺序表递增有序，插入元素 x，仍递增有序 【查+增】
- 6、用顺序表最后一个元素覆盖整个顺序表中最小元素（若有多个则选取第一个），并返回该最小元素。若最后一个就是最小元素时，则不改变顺序表的状态。仅返回这个最小元素即可。 【查+改】
- 7、删除顺序表中第一个值为 x 的元素 【查+删一】
- 8、删除顺序表中所有值为 x 的元素（拓展：从顺序表中删除给定值在 s 到 t 之间（包含 s 和 t）的所有元素） 【查+删多】

链表：

- 1、分别采用头插法和尾插法建立一个带头结点的单链表（思考：如何创建一个不带头结点的单链表，强化篇讲解） 【头插和尾插】
- 2、一个带头结点递增有序的单链表 L，申请一个值为 x 的结点空间，将其插入 L 后，单链表仍保持递增有序 【查+增（增是创建新结点后再插）】
将带头结点的单链表就地逆置，保证空间复杂度为 $O(1)$ 【经典头插】
- 3、删除带头结点单链表中第一个值为 x 的结点 【查+删一】
- 4、删除带头结点单链表中所有值为 x 的结点（拓展：若删除给值在 s 到 t 之间（不包含 s 和 t）的所有结点呢？） 【查+删多】
- 5、试编写算法将带头结点的单链表就地逆置，即不需要借助辅助空间，保证空间复杂度为 $O(1)$ 【头插法逆置，十分重要!!!】
- 6、试编写在带头结点的单链表 L 中删除最小值点的算法 【查+删一】
- 7、递增有序地输出单链表中的各结点的数值，并释放结点空间 【查+删多】

- 8、将一个带头节点单链表值最小的结点移动到链表的最前面 【查+插】
- 9、设有一个由正整数组成的无序单链表，实现以下功能： 【查+插+删一】
 - 1、找出最小值结点(非最后一个且唯一)
 - 2、若该数值是奇数，将其于后继结点交换（注意不是数值交换）
 - 3、若该数值是偶数，则将其后继结点删除
- 10、分别用头插法和尾插法创建一个带头节点的双向链表 【对比第 1 题】
- 11、将带头节点双向链表中值最小的结点移动到链表最前端 【对比第 8 题】
- 12、设有一个带头结点的循环单链表，其结点值为正整数，设计算法反复找出链表内最小值并不断输出，并将结点从链表中删除，直到链表为空，再删除表头结点 【对比第 7 题】

栈和队列：

- 1、栈的基本操作 【初始化、入、出、判空、取栈顶】
- 2、判断单链表中全部 n 个字符是否回文 【栈的经典应用】
- 3、判断一个表达式中圆括号是否配对（拓展：若还有花括号多种类型的括号呢？） 【栈的经典应用】
- 4、假设一个序列为 HSSHHS，运用栈的知识，编写算法将 S 全部提到 H 之前，即为 SSSHHHH
- 5、两个栈 s1,s2 采用顺序存储，共享一个存储区 $[0, \dots, \text{maxsize}-1]$ 。采用栈顶相向，迎面增长的存储方式，设计 s1,s2 入栈和出栈的操作。 【共享栈】
- 6、队列的基本操作 【初始化、入、出、判空】
- 7、设以带头节点的循环单链表表示队列，只设有队尾指针。请写出入队、出队的算法，复杂度要求均为 $O(1)$ 。 【链表模拟队列】
- 8、利用两个栈 s1 和 s2 来模拟一个队列。如何利用栈的运算实现该队列的三个运算：入队、出队和判断队列为空。 【栈模拟队列】
- 9、判断字符串是否回文 【注意与链式存储判断回文的区别】

- 10、判断子串 s_2 是否匹配母串 s_1 ，若匹配，输出匹配到的第一个字符所在索引。否则输出 -1。【顺序存储字符串暴力匹配】
- 11、有两个链表 A 和 B，判断 B 是否为 A 的连续子序列【链式存储匹配】

树（基础篇都是很简单的题目，一套递归公式全部秒杀）

- 1、使用先序中序后序递归遍历二叉树
 - 2、计算二叉树中所有结点个数
 - 3、计算二叉树中所有叶子结点的个数
- 拓展：计算二叉树中所有双分支的结点个数
- 4、求二叉树中值为 x 的层号
 - 5、计算二叉树的最大深度（高度）
 - 6、找出二叉树中值最大的结点
 - 7、查找二叉树中 data 域等于 key 的结点是否存在，若存在，将 q 指向它，否则 q 为空
 - 8、输出先序遍历第 k 个结点的值
 - 9、把二叉树所有结点左右子树交换
 - 10、判断二叉树是不是正则二叉树（即每个结点的度均为 0 或 2）
 - 11、先序非递归遍历二叉树【牢记模板，学会套用，中、后同理】
 - 12、中序非递归遍历二叉树
 - 13、后序非递归遍历二叉树
 - 14、层次遍历二叉树

图

- 1、分别使用邻接表和邻接矩阵创建一个图
- 2、邻接表实现图的广度优先遍历（BFS）
- 3、邻接矩阵实现图的广度优先遍历（BFS）
- 4、设计算法，求无向连通图距顶点 v 最远的一个结点（即路径长度最大）
- 5、邻接表实现图的深度优先遍历（DFS）

- 6、邻接矩阵实现图的深度优先遍历（DFS）
- 7、有向图采用邻接表存储，判断顶点 V_i 和顶点 V_j 之间是否存在路径
拓展：有向图采用邻接矩阵存储呢？（经典有无路径问题，DFS/BFS）
- 8、在有向图中，如果顶点 r 到图中所有顶点都存在路径，则称 r 为图的根结点。编写代码输出有向图中所有根结点。（第 7 题的应用）
- 9、求无向图的连通分量个数（DFS/BFS）

查找

- 1、在有序表中二分查找值为 key 的元素【二分查找，递归和非递归解法】
- 2、判断给定二叉树是否是二叉(搜索)排序树【树递归的应用】
- 3、寻找二叉排序树中最大值和最小值结点【二叉排序树的应用】
- 4、求出值为 key 的结点在二叉排序树的层次【二叉排序树的应用】

排序

- 1、直接插入排序
- 2、折半插入排序
- 3、选择排序
- 4、冒泡排序
- 5、快速排序

顺序表默认结构体:

1、在顺序表 L 的第 k 个位置插入元素 x (增)

2、删除顺序表 L 的第 k 个元素并返回其值 (删)

3、将顺序表中的元素逆置 (改)

4、查找顺序表中第一个值为 x 的元素的位置(查)

5、顺序表递增有序，插入元素 x ，仍递增有序

6、用顺序表最后一个元素覆盖整个顺序表中最小元素（若有多个则选取第一个），并返回该最小元素。若最后一个就是最小元素时，则不改变顺序表的状态。仅返回这个最小元素即可。

7、删除顺序表中第一个值为 x 的元素

8、删除顺序表中所有值为 x 的元素

单链表默认结构体:

1、分别采用头插法和尾插法建立一个带头结点的单链表

2、一个带头结点递增有序的单链表 L ，申请一个值为 x 的结点空间，将其插入 L 后，单链表仍保持递增有序

3、删除单链表中第一个值为 x 的结点

4、删除单链表中所有值为 x 的结点

5、试编写算法将带头结点的单链表就地逆置，即不需要借助辅助空间，保证空间复杂度为 $O(1)$

6、试编写在带头结点的单链表 L 中删除最小值点的高效算法（已知最小值唯一）

7、给定一个单链表，按递增有序地输出单链表中各结点的数值，并释放结点所占空间（不断寻找最小值）

8、将一个带头节点单链表中值最小的结点移动到整个链表的最前面。

9、设有一个由正数组成的无序单链表，编写程序之下以下功能：

- 1、找出最小值结点(非最后一个且唯一)
- 2、若该数值是奇数，将其于后继结点的结点交换（注意不是数值交换）
- 3、若该数值是偶数，则将其后继结点删除

10、分别采用头插法和尾插法创建一个带头节点的双向链表

11、将一个带头节点双向链表中值最小的结点移动到整个链表的最前面。

12、设有一个带头结点的循环单链表，其结点值为正整数，设计算法反复找出链表内最小值并不断输出，并将结点从链表中删除，直到链表为空，再删除表头结点

栈的结构体:

队列的结构体

1、栈的基本操作

2、判断单链表中全部 n 个字符是否回文

3、判断一个表达式中圆括号是否配对

4、假设一个序列为 **HSSHHS**，运用栈的知识，编写算法将 **S** 全部提到 **H** 之前，即为 **SSSHHH**

5、两个栈 s_1, s_2 都采用顺序存储，并共享一个存储区 $[0, \dots, \text{maxsize}-1]$ 。采用栈顶相向，迎面增长的存储方式，设计 s_1, s_2 入栈和出栈的操作。

6、队列的基本操作

7、设以带头节点的循环单链表表示队列，只设有队尾指针。请写出入队、出队的算法，复杂度要求均为 $O(1)$ 。

8、用两个栈 **s1** 和 **s2** 来模拟一个队列。如何利用栈的运算实现该队列的两个运算：入队和出队。

9、判断字符串是否回文

10、判断子串 **s2** 是否匹配母串 **s1**，若匹配，输出匹配到的第一个字符所在索引。否则输出-1。

11、有两个链表 A 和 B，判断 B 是否是 A 的连续子序列

二叉树的结构体 (链式存储) (二叉链表存储)

2、计算二叉树中所有结点个数

1、使用先序中序后序递归遍历二叉树

3、计算二叉树中所有叶子结点的个数

计算型：

拓展：如何计算二叉树中所有双分支的结点个数

4、求二叉树中值为 x 的层号

5、计算二叉树的最大深度（高度）

6、找出二叉树中值最大的结点

7、查找二叉树中 **data** 域等于 **key** 的结点是否存在，若存在，将 **q** 指向它，否则 **q** 为空

8、输出先序遍历第 **k** 个结点的值

9、把二叉树所有结点左右子树交换

10、判断二叉树是不是正则二叉树（即每个结点的度均为 0 或 2）

11、先序非递归遍历二叉树

需要讲义对应讲解视频，可加下方微信获取



12、中序非递归遍历二叉树

13、后序非递归遍历二叉树

14、层次遍历二叉树

1、分别使用邻接表和邻接矩阵创建一个图

2、邻接表实现图的广度优先遍历（BFS）

3、邻接矩阵实现图的广度优先遍历 (BFS)

4、设计算法，求无向连通图距顶点 v 最远的一个结点 (即路径长度最大)

5、邻接表实现图的深度优先遍历 (DFS)

6、邻接矩阵实现图的深度优先遍历 (DFS)

7、有向图采用邻接表存储，设计算法判断顶点 V_i 和顶点 V_j 之间是否存在路径

8、在有向图中，如果顶点 r 到图中所有顶点都存在路径，则称 r 为图的根结点。编写代码输出有向图中所有根结点。

9、求无向图的连通分量个数

1、在有序表中二分查找值为 **key** 的元素

2、判断给定二叉树是否是二叉(搜索)排序树

3、寻找二叉排序树中最大值和最小值结点

4、求出值为 **key** 的结点在二叉排序树的层次

1、直接插入排序

2、折半插入排序

3、选择排序

4、冒泡排序

5、快速排序