



# 数据结构强化课考试

注1：本次考试时间共120分钟（408考试中，平均每个大题可分配20分钟左右的做题时间，因此该试卷共包含6个数据结构大题）

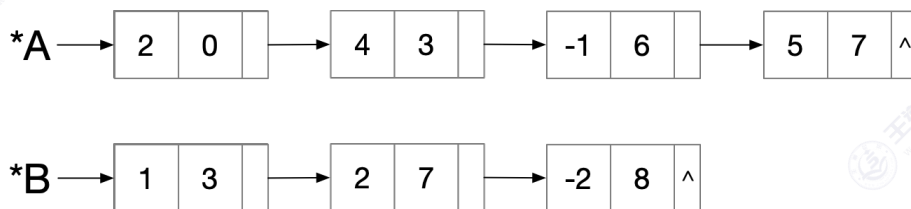
注2：本试卷的个别题目难度略高于真题

## 算法题

一、数学上的一元多项式  $P(x) = p_0 x^0 + p_1 x^1 + p_2 x^2 + \dots + p_n x^n$ ，可以用一个单链表来存储，结点的数据结构定义如下：

```
typedef struct Node {  
    float co;    //系数  
    int ex;      //指数  
    struct Node *next;    //指向下一个结点  
} *Polynomial;
```

其中，co 表示每一项的“系数”，ex 表示每一项的“指数”，next 为指向下一个结点的指针。我们规定：“系数”为0的项无需存储；各个项在链表中按“指数”递增存放；单链表没有头结点。例如，多项式  $A(x) = 2 + 4x^3 - x^6 + 5x^7$  和 多项式  $B(x) = x^3 + 2x^7 - 2x^8$  可表示为：



现要求设计一个尽可能高效的算法，实现两个多项式的加法，并返回相加之后的结果。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。

二、已知一棵非空二叉树 T 高度为h，结点总数为n，采用二叉链表存储结构，结点的数据结构定义如下：

```
//二叉树的结点（链式存储）  
typedef struct BiTNode{  
    char data;    //数据域  
    struct BiTNode *lchild, *rchild;    //左、右孩子指针  
}BiTNode, *BiTree;
```

请设计一个算法，求树 T 的宽度（即具有结点数最多的那一层的结点个数），要求：



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

三、一个长度为 $n$ 的升序整数序列 $S$ 中，只有常数 $K$ （ $K$ 的值已知）出现了若干次，其他数最多都只出现一次。试设计一个在时间和空间两方面都尽可能高效的算法，返回  $K$  出现的次数。

- (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。

## 应用题

四、请回答以下问题：

- (1) 队列在顺序存储时的“假溢出”现象指什么？
- (2) 简述一种可行的假溢出的解决方法。
- (3) 若用数组 $q[1...m]$ 表示队列，队列头指针 $front$ 、尾指针 $rear$ 的初值均为1，基于（2）中的方法，如何求队列的当前长度？如何判定队空？如何判定队满？

【北京邮电大学803 2018年】

五、将关键字序列 {116, 100, 101, 115, 117, 103} 依次插入到初始为空的平衡二叉树（AVL树），给出每插入一个关键字后的平衡树，并说明其中可能包含的平衡调整步骤（即：先说明是哪个结点失去平衡，然后说明做了什么平衡处理）；然后分别给出前序、中序和后序遍历该二叉树的输出结果。【中国科学院大学863 2019年】

六、设一组有序的记录关键字序列为 (14, 19, 25, 36, 48, 51, 63, 84, 91)，运用二分法进行查找，请给出二分查找的判定树，以及查找关键字 84 时的比较次数，并计算出查找成功时的平均查找长度。【华中科技大学834 2019年】