

数据结构与算法

期中试卷解析（2）

简答题

1、什么是数据结构？什么是算法？他们有什么联系？

答：数据结构是相互之间存在一种或多种特定关系的数据元素的集合。

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每一条指令表示一个或多个操作。

数据结构是对数据的描述，算法是对数据的操作。

2、请分析比较顺序表与链表的特点，并说明它们的适用场合。

答：顺序表是用一片连续的空间存储线性表，而链表可以不用连续的空间存储线性表，元素之间通过指针相互连接。顺序表用在不经常插入或删除操作的线性表，而链表方便支持插入和删除操作。

3、栈与队列有什么区别和联系？各有什么特点？

答：栈是一种特殊的线性表，它要求插入和删除操作都在同一端进行。队列也是一种特殊的线性表，它规定插入和删除操作必须在不同端进行。

栈的特点是先进后出，队列的特点是先进先出。

简答题

4、什么是广义表？请简述广义表与线性表的区别？

答：广义表是一种特殊的线性表，它的元素本身可以是广义表。而线性表要求其元素是不可再分的单元。

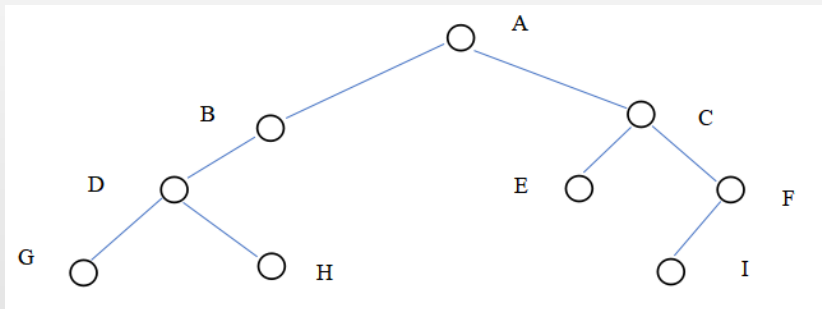
5、设一棵度为k的树中有 n_1 个度为1的结点， n_2 个度为2的结点，... ..， n_k 个度为k的结点，试问该树有多少个叶子结点？

答：设该树有m条边，则 $m = \sum_{i=1}^k i \cdot n_i$ ($i=1,2,\dots,k$)，设该树有 n_0 个叶子节点，则该树共有n个节点， $n = \sum_{j=0}^k n_j$ ($j=0,1,2,\dots,k$)，因为 $n = m + 1$ ，故有 $\sum_{j=0}^k n_j = \sum_{i=1}^k i \cdot n_i + 1$ ($i=1,2,\dots,k; j=0,1,2,\dots,k$)，可得

$$n_0 = 1 + \sum_{i=2}^k [(i-1) \cdot n_i], i=2,\dots,k$$

简答题

6、已知一棵二叉树的先序遍历序列和中序遍历序列分别为ABDGHCEFI和GDHBAECIF，请画出这棵二叉树，然后给出该树的后序遍历序列。



答：后序遍历序列：GHDBEIFCA

二、算法题

1、设有两个集合A和集合B，要求设计生成集合 $C=A \cap B$ 的算法，其中集合A、B和C用链式存储结构表示。

```
typedef struct node
```

```
{
```

```
    int data;
```

```
    struct node *next;
```

```
}lklist;
```

```
void intersection(lklist *ha,lklist
```

```
*hb,lklist *&hc)
```

```
{
```

```
    lklist *p,*q,*t;
```

```
for(p=ha,hc=0;p!=0;p=p->next)
```

```
{
```

```
    for(q=hb;q!=0;q=q->next)
```

```
        if (q->data==p->data) break;
```

```
    if(q!=0)
```

```
    {
```

```
t=(lklist*)malloc(sizeof(lklist));
```

```
t->data=p->data;t->next=hc;
```

```
hc=t;
```

```
}
```

```
}
```

```
}
```

二、算法题

2、设二叉树以二叉链表形式存储，请编写一个求叶子结点总数的算法。

```
Typedef Struct Node
```

```
{  DataType: data;
```

```
Struct Node *Lchild;
```

```
Struct Node *Rchild;
```

```
} *BiTree;
```

```
int TreeOper(BiTree T)
```

```
{
```

```
    if(!T) return 0;  //若T为空， 返还0
```

```
    if(!T->Lchild && !T->Rchild) return 1;
```

```
    //若没有子节点， 则T为一个叶子结点
```

```
    return TreeOper (T->Lchild)+ TreeOper (T->Rchild); //返还左子树中叶子结点数与右子树中叶子结点数之和
```

```
}
```

二、算法题(常见问题)

①指针操作错误——请好好练习c/c++指针运算

```
while(p)
```

```
{  
    while(q)  
    {  
        q=q->next;  
        if(q->data==p->data)  
            // 这样的操作如果没有别的退出循环的条件，那必将以访问空指针出错退出程序  
            {  
                .....  
            }  
    }  
}
```

②其他编程语言语法问题

循环语句使用、变量定义、逻辑运算符使用等等

二、算法题(常见问题)

③先排序，再清理——一次能做好的事情，用两倍的时间来做，是为了情怀吗？

④用辅助数组——如何保证数组不越界？

⑤基本概念理解——空链表是无辜的、链表头结点不是必须的。

⑥典型递归算法编写不熟悉。

⑦概念理解不够全面——空树也是树、空链表也是链表。