

一、 一、 单选题（每小题 2 分，共 8 分）

1、1、在一个长度为 n 的顺序线性表中顺序查找值为 x 的元素时，查找成功时的平均查找长度（即 x 与元素的平均比较次数，假定查找每个元素的概率都相等）为（ ）。

- A n B $n/2$ C $(n+1)/2$ D $(n-1)/2$

2、2、在一个单链表中,若 q 所指结点是 p 所指结点的前驱结点,若在 q 与 p 之间插入一个 s 所指的结点,则执行()。

- A $s \rightarrow \text{link} = p \rightarrow \text{link}; \quad p \rightarrow \text{link} = s;$ B $p \rightarrow \text{link} = s; \quad s \rightarrow \text{link} = q;$
C $p \rightarrow \text{link} = s \rightarrow \text{link}; \quad s \rightarrow \text{link} = p;$ D $q \rightarrow \text{link} = s; \quad s \rightarrow \text{link} = p;$

3、3、 栈的插入和删除操作在（ ）进行。

- A 栈顶 B 栈底 C 任意位置 D 指定位置

4、4、 由权值分别为 11, 8, 6, 2, 5 的叶子结点生成一棵哈夫曼树，它的带权路径长度为（ ）

- A 24 B 71 C 48 D 53

二、 二、 填空题（每空 1 分，共 32 分）

1、1、数据的逻辑结构被分为_____、_____、_____和_____四种。

2、2、一种抽象数据类型包括_____和_____两个部分。

3、3、在下面的数组 a 中链接存储着一个线性表，表头指针为 $a[o].\text{next}$ ，则该线性表为_____。

a	0	1	2	3	4	5	6	7	8	
	60	56	42	38		74	25			data next
4	3	7	6	2		0	1			

4、4、在以 HL 为表头指针的带表头附加结点的单链表和循环单链表中，判断链表为空的条件分别为_____和_____。

5、5、用具有 n 个元素的一维数组存储一个循环队列，则其队首指针总是指向队首元素的_____，该循环队列的最大长度为_____。

6、6、当堆栈采用顺序存储结构时，栈顶元素的值可用_____表示；当堆栈采用链接存储结构时，栈顶元素的值可用_____表示。

7、7、一棵高度为 5 的二叉树中最少含有_____个结点，最多含有_____个结点；

一棵高度为 5 的理想平衡树中，最少含有_____个结点，最多含有_____个结点。

8、8、在图的邻接表中，每个结点被称为_____，通常它包含三个域：一是_____；二是_____；三是_____。

9、9、在一个索引文件的索引表中，每个索引项包含对应记录的_____

和_____两项数据。

10、 10、 假定一棵树的广义表表示为 $A(B(C, D(E, F, G), H(I, J)))$ ，则树中所含的结点数为_____个，树的深度为_____，树的度为_____，结点 H 的双亲结点为_____，孩子结点为_____。

11、 11、 在堆排序的过程中，对任一分支结点进行筛运算的时间复杂度为_____，整个堆排序过程的时间复杂度为_____。

12、 12、 在对 m 阶的 B_树插入元素的过程中，每向一个结点插入一个索引项（叶子结点中的索引项为关键字和空指针）后，若该结点的索引项数等于_____个，则必须把它分裂为_____个结点。

三、 三、 运算题（每小题 6 分，共 24 分）

1、 1、 已知一组记录的排序码为 (46, 79, 56, 38, 40, 80, 95, 24)，写出对其进行快速排序的每一次划分结果。

2、 2、 一个线性表为 $B = (12, 23, 45, 57, 20, 03, 78, 31, 15, 36)$ ，设散列表为 $HT[0..12]$ ，散列函数为 $H(key) = key \% 13$ 并用线性探查法解决冲突，请画出散列表，并计算等概率情况下查找成功的平均查找长度。

3、 3、 已知一棵二叉树的前序遍历的结果序列是 ABECKFGHIJ，中序遍历的结果是 EBCDAFHIGJ，试写出这棵二叉树的后序遍历结果。

4、 4、 已知一个图的顶点集 V 各边集 G 如下：

$V = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$;

$E = \{(0, 1), (0, 4), (1, 2), (1, 7), (2, 8), (3, 4), (3, 8), (5, 6), (5, 8), (5, 9), (6, 7), (7, 8), (8, 9)\}$

当它用邻接矩阵表示和邻接表表示时，分别写出从顶点 V_0 出发按深度优先搜索遍历得到的顶点序列和按广度优先搜索遍历得到的顶点序列。

假定每个顶点邻接表中的结点是按顶点序号从大到小的次序链接的。

图	深度优先序列	广度优先序列
邻接矩阵表示时		
邻接表表示时		

四、 四、 阅读算法，回答问题（每小题 8 分，共 16 分）

1、假定从键盘上输入一批整数，依次为：78 63 45 30 91 34 -1，请写出输出结果。

```
#include <iostream.h>
#include <stdlib.h>
const int stackmaxsize = 30;
typedef int elemtype;
struct stack {
    elemtype stack [stackmaxsize];
    int top;
};
#include "stack.h"
Void main ( )
{
    stack a;
    initstack(a);
    int x;
    cin >>x;
    while (x!= -1) {
        push (a, x );
        cin >>x;
    }
    while (!stackempty (a))
        cout <<pop (a) <<" " ;
    cout <<endl;
}
```

该算法的输出结果为：

2、阅读以下二叉树操作算法，指出该算法的功能。

```
Template <calss type> void BinTree <Type> ::
unknown (BinTreeNode<Type>*t) {
    BinTreeNode< Type> *p =t, *temp;
    if (p!=NULL) {
        temp = p->leftchild;
        p->leftchild = p->rightchild;
        p->rightchild = temp;
        unknown(p->leftchild);
    }
}
```

```

        undnown(p→rightchild);
    }
}

```

该算法的功能是： _____

五、 五、 算法填空，在画有横线的地方填写合适的内容（10 分）

对顺序存储的有序表进行二分查找的递归算法 。

```

int Binsch( ElemType A[ ],int low ,int high,KeyType K )
{
    if (low <= high)
    {
        int mid = 1
        if ( K== A[ mid ].key )
            return mid;
        else if ( K < A[mid].key)
            return 2
        else
            return 3
    }
    else
        return 4
}

```

六、 六、 编写算法（10 分）

编写算法，将一个结点类型为 Lnode 的单链表按逆序链接，即若原单链表中存储元素的次序为 a_1, \dots, a_{n-1}, a_n ，则逆序链接后变为, a_n, a_{n-1}, \dots, a_1 。

Void contrary (Lnode * & HL)

数据结构试题（答案）

一、单选题（每小题 2 分，共 8 分）

题 号	1	2	3	4
答 案	C	D	A	B

二、填空题（每空 1 分，共 32 分）

- 1: 集合、线性、树、图;
- 2: 数据描述、操作声名;
- 3: (38, 56, 25, 60, 42, 74);

- 4: HL→next=NULL; HL=HL→next;
- 5: 前一个位置; n-1;
- 6: S.stack[S.top]; HS→data;
- 7: 5 31
- 8: 边结点、邻接点域、权域、链域;
- 9: 索引值域、开始位置域;
- 10: 10、3、3、B、I和J;
- 11: $O(\log_2 n)$ 、 $O(n \log_2 n)$;
- 12: m、m-1

三、运算题（每小题6分，共24分）

1、

划分次序	划分结果
第一次	[38 24 40] 46 [56 80 95 79]
第二次	24 [38 40] 46 [56 80 95 79]
第三次	24 38 40 46 [56 80 95 79]
第四次	24 38 40 46 56 [80 95 79]
第五次	24 38 40 46 56 79 [80 95]
第六次	24 38 40 46 56 79 80 95

2、

0 1 2 3 4 5 6 7 8 9 10 11 12

78		15	03		57	45	20	31		23	36	12
----	--	----	----	--	----	----	----	----	--	----	----	----

查找成功的平均查找长度： $ASL_{succ} = 14/10 = 1.4$

3、此二叉树的后序遍历结果是：EDCBIHJGFA

4、

图	深度优先序列	广度优先序列
邻接矩阵表示时	0, 1, 2, 8, 3, 4, 5, 6, 7, 9	0, 1, 4, 2, 7, 3, 8, 6, 5, 9
邻接表表示时	0, 4, 3, 8, 9, 5, 6, 7, 1, 2	0, 4, 1, 3, 7, 2, 8, 6, 9, 5

四、阅读算法，回答问题（每小题8分，共16分）

- 1、1、该算法的输入结果是：34 91 30 45 63 78
- 2、2、该算法的功能是：交换二叉树的左右子树的递归算法。

五、算法填空，在画有横线的地方填写合适的内容（10分）

- 1、1 是： $(low + high) / 2$;
- 2 是：Binsch(A,low,mid - 1,K);
- 3 是：Binsch(A,mid+1,high,K);
- 4 是：-1;

六、编写算法（10分）

根据编程情况，酌情给分。

```
{
    Lnode *P=HL;
    HL=NULL;
    While (p!=null)
    {
```

```
Lnode*q=p;  
P=p→next;  
q→next=HL;  
HL=q;
```