

试卷五

一、填空题 (共 17 分, 每空 1 分)

1. 在数据结构中, 数据元素之间通常有下列四类基本结构: _____、_____, _____和_____; 有两种物理结构 (存储结构), 分别 _____、_____。
2. n 个顶点的连通图至少有 _____ 条边; 任何一个具有 n 个结点的完全无向图有 _____ 条边; n 个结点的完全有向图有 _____ 条弧。
3. 在无向图 G 的邻接矩阵 A 中, 若 $A[i][j]$ 等于 1, 则 $A[j][i]$ 等于_____。
4. 通过建立 Hash 表查找元素, 理想情况下, 查找元素的时间复杂度为_____。
5. 长度为 11 的有序序列: 1 12 13 24 35 36 47 58 59 69 71 进行等概率查找, 如果采用顺序查找, 则平均查找长度为_____, 如果采用二分查找, 则平均查找长度为_____, 如果采用哈希查找, 哈希表长为 15, 哈希函数为 $H(\text{key}) = \text{key} \% 13$, 采用线性探查解决地址冲突, 即 $\text{di} = (H(\text{key}) + i) \% 15$, 则平均查找长度为 (保留 1 位小数) _____。
6. 通过衡量一个算法的 _____ 复杂度和 _____ 复杂度来判定一个算法的好坏。
7. 将下三角矩阵 $A[8,8]$ 的下三角部分逐行地存储到起始地址为 1000H 的内存单元中 (下标从 0 开始, 不存储上三角部分), 已知每个元素占 4 个单元, 则 $A[5,4]$ 的地址是 (要求十六进制数) _____。

二、选择题 (共 13 分, 每题 1 分)

1. 下面带有 @ 标记的语句的频度 ($n > 10$) 是 []

```
for(int i=0; i<n-1; i++)  
    for(int j=i+1; j<n; j++)  
        @cout<<i<<j<<endl;
```

A $n*(n-1)/2$ B $n*n/2$ C $n*(n+1)/2$ D 不确定
2. 已知使用顺序表存储数据, 表长为 n , 假设在表中的任意位置插入元素的概率相等, 则插入一个元素, 平均需要移动的元素个数 []

A $(n-1)/2$ B $n/2$ C $(n+1)/2$ D 不确定
3. 在双向链表 p 所指结点之后插入 s 所指结点的操作是 []

- A. $p \rightarrow \text{right} = s$; $s \rightarrow \text{left} = p$; $p \rightarrow \text{right} \rightarrow \text{left} = s$; $s \rightarrow \text{right} = p \rightarrow \text{right}$;
 B. $p \rightarrow \text{right} = s$; $p \rightarrow \text{right} \rightarrow \text{left} = s$; $s \rightarrow \text{left} = p$; $s \rightarrow \text{right} = p \rightarrow \text{right}$;
 C. $s \rightarrow \text{left} = p$; $s \rightarrow \text{right} = p \rightarrow \text{right}$; $p \rightarrow \text{right} = s$; $p \rightarrow \text{right} \rightarrow \text{left} = s$;
 D. $s \rightarrow \text{left} = p$; $s \rightarrow \text{right} = p \rightarrow \text{right}$; $p \rightarrow \text{right} \rightarrow \text{left} = s$; $p \rightarrow \text{right} = s$;
- 4、字符串相等的充分必要条件是[]
 A. 串长度相等 B. 串使用相同的存储结构
 C. 串相同位置对应的字符相等 D. A 和 C
- 5、将一个递归算法改为对应的非递归算法时，通常需要使用[]
 A. 数组 B. 栈 C. 队列 D. 二叉树
- 6、一个栈的入栈序列 1, 2, 3, 4, 5, 则栈的不可能的输出序列是[]
 A. 12345 B. 54321 C. 32514 D. 12354
- 7、设循环队列中数组的下标范围是 $1 \sim n$, 其头尾指针分别为 f 和 r , 则其元素个数为
 []
 A. $r-f$ B. $r-f+1$ C. $(r-f) \bmod n + 1$ D. $(r-f+n) \bmod n$
- 8、已知图 G , 求从图中的一个顶点到其他顶点的最短路径, 一般使用的算法是[]
 A. 普里姆(Prim)算法 B. 克鲁斯卡尔(Kruskal)算法
 C. 迪杰斯特拉(Dijkstra)算法 D. 弗洛伊德(Floyd)算法
- 9、某二叉树的前序遍历结点访问顺序是 ABDEF CGH, 中序遍历的结点访问顺序是 DBFEAGHC, 则其后序遍历的结点访问顺序是[]
 A. DFEBHCGA B. DFEBHGCA
 C. DEFBHGCA D. DFEHBGCA
- 10、正则二叉树是只有度为 0 和 2 的结点的二叉树, 已知正则二叉树的叶子结点个数为 n , 则该二叉树总得结点数为[]
 A. $n+1$ B. $2*n$ C. $2*n+1$ D. $2*n-1$
- 11、下面关于排序的说法错误的是[]
 A 快速排序、归并排序都是一种不稳定的排序方法
 B 直接插入排序和折半插入排序移动元素的次数相同
 C 简单选择排序移动元素的次数最少
 D 根据排序需要的平均时间, 快速排序是目前最好的一种内部排序方法
- 12、折半查找有序表 (3, 4, 5, 10, 13, 14, 20, 30), 若查找元素 3, 则被比较的元素依次为[]

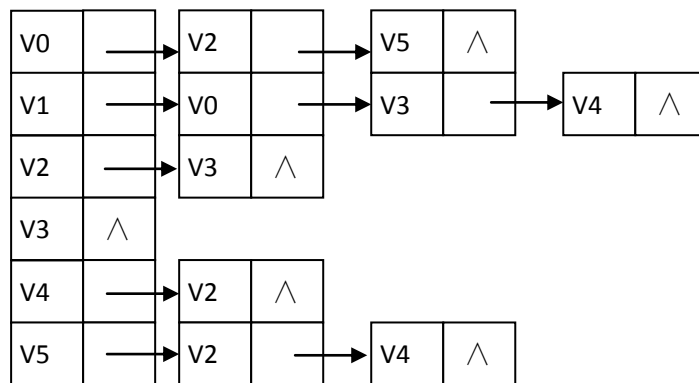
A.10,20,30 B.10,14,30 C.13,3 D.10, 4, 3

13、下面关于栈和队列的说法正确的是[]

- A. 栈是先进先出的线性表，队列是后进先出的线性表
- B. 栈是先进先出的线性表，队列也是先进先出的线性表
- C. 栈是后进先出的线性表，队列是先进先出的线性表
- D. 栈是后进先出的线性表，队列也是后进先出的线性表

三、简答题 （共 22 分）

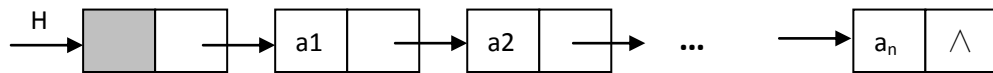
1、（9 分）已知邻接表（如下图所示），画出其对应的有向图 G，并写出从 V0 开始深度优先搜索和广度优先搜索的序列



2、（4 分）根据下面的三元组，写出相应的稀疏矩阵（矩阵为 6 行 7 列）

i	j	k
1	2	12
1	3	9
2	5	5
3	1	-3
3	6	14
4	3	13
5	2	18
6	1	15
6	7	8

3、(9分) 已知 L 是带头结点的单链表，表头指针为 H (如下图所示)：

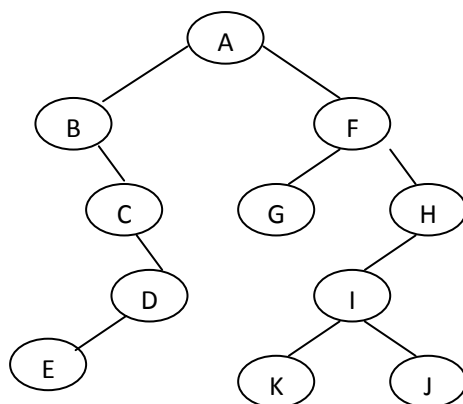


- (1) 如果在表头插入一个结点 (s 为指向该结点的指针)，则相应的代码是 _____；
- (2) 如果在表头删除一个结点，则相应的代码是 _____；
- (3) 如果在表尾插入一个结点 (s 为指向该结点的指针)，则相应的代码是 _____；

- A while (p!=NULL) p=p->next;
- B s->next = H->next;
- C p->next = s;
- D Node *p = H->next;
- E s->next = p->next;
- F H->next = s;
- G H->next = p->next;
- H while (p->next!=NULL) p=p->next;
- I p->next = NULL;
- J Node *p = H;
- K delete p;
- L delete H;

四、综合题 (共 34 分)

1、(10分) 根据下面的二叉树，写出先序遍历、中序遍历、后序遍历的序列，再将该二叉树转化成森林



2、(14分) 已知某系统在通信联络中只可能出现八种字符，其概率分别为 0.07(A)、0.19(B)、

0.02(C)、0.06(D)、0.32(E)、0.03(F)、0.21(G)、0.10(H)

- (1) 画出哈夫曼树，并计算其带权路径长度 WPL;
- (2) 最长的编码为几位，对应哪些字符？最短的编码为几位，对应哪些字符；
- (3) 如果电文是 ABCDEFGH，压缩前每个电文使用 8bit ASCII 编码，则采用上面的 Huffman 编码，其压缩比是多少？

3、(10 分) 已知序列 (19, 49, 55, 32, 66, 26, 108, 58, 46, 95, 31) 判断是否为小 (顶) 根堆？

- (1) 若否，则以最少的移动次数将它们调整为小 (顶) 根堆
- (2) 写出一趟堆排序的结果，即输出堆顶元素(将该元素交换到最后位置)，并调整成新堆的结果
(要求画出最后的堆结构和线性序列)

五、编程填空 (共 14 分，每空 2 分)

1、直接插入排序 (升序)

说明：待排序记录为整型，存放在数组 list 中，其中 list[0]留空。

```
void InsertSort(int list[], int len)
{
    for(int _____; i<=len; i++)
        if( list[i]<list[i-1])
        {
            list[0]=_____
            list[i] = list[i-1];
            for(int j=i-2; list[0]<list[j]; j--)
                _____; //记录后移
            _____;
        }
}
```

2、简单选择排序 (升序)

说明：待排序记录为整型，存放在数组 list 中，其中 list[0]留空。

```

void SelectSort(int list[], int len)
{
    int j;
    for(int i=1; i<=len; i++)
    {
        j=i;
        for(int k=i+1; k<=len; k++)    //选取最小记录的位置
            if (list[j]>list[k])
                _____;
        if(_____)
        {
            list[0]=list[i];  list[i]=list[j];  _____;    //交换数据
        }
    }
}

```