

题号	一	二	三	四	五	六	七	八	九	总分	签名
得分											

本试卷中涉及的链表、循环链表、顺序表及二叉链表的类型定义如下：

链表、循环链表定义如下：

```
typedef struct Node{
    int data;           // 数据域
    struct Node *next;  // 指针域
}LNode, *LinkList;
```

顺序表的类型定义如下：

```
typedef struct {
    int *elem;          //存储空间基址
    int length;         //当前长度
    int listsize;       //当前分配的存储容量,以一数据元素存储长度为单位
}SqList;
```

二叉链表定义如下：

```
typedef struct bitnode{    //定义二叉树节点结构
    char data;             //数据域
    struct bitnode *lchild,*rchild; //左右孩子指针域
}BiTNode, *BiTree;
```

一、填空（15 分，每空 1 分）

1. 数据结构包括数据的 逻辑结构、数据的 存储结构 和数据的运算这三个方面的内容。
2. 当线性表的元素总数基本稳定，且很少进行插入和删除操作，但要求以最快的速度存取线性表中的元素时，应采用 顺序 存储结构。
3. 在单链表中，要删除某一个指定的结点，必须找到该结点的 前驱 结点。
4. 一个栈的输入序列为 1, 2, 3, ..., n, 若输出序列的第一个元素是 n, 输出第 i ($1 \leq i \leq n$) 个元素是 $n+1-i$ 。
5. n 个顶点的连通无向图用邻接矩阵表示时，该矩阵至少有 $2(n-1)$ 个非零元素。
6. 图的 深度优先搜索 搜索方法类似于二叉树的先根遍历。

7. 在一棵二叉树中,度为零的结点的个数为 N_0 , 度为 2 的结点的个数为 N_2 , 则有 $N_0 = N_2 + 1$ 。
8. 排序时扫描待排序记录序列, 依次比较相邻的两个元素的大小, 逆序时就交换位置。这是 冒泡 排序方法的基本思想。
9. 一棵高度为 4 的二叉树中最多含有 15 个结点。
10. 设线性表中有 n 个数据元素, 则在链式存储结构上实现顺序查找的平均时间复杂度为 $O(n)$ 。
11. 若用 n 表示图中的顶点数目, 则有 $n(n-1)/2$ 条边的无向图被称为完全图。
12. 设一个连通图 G 中有 n 个顶点 e 条边, 则其最小生成树上有 $n-1$ 条边。
13. 有向图的边集为 $\{ \langle a, c \rangle, \langle a, e \rangle, \langle e, b \rangle, \langle e, d \rangle, \langle b, d \rangle, \langle d, c \rangle, \langle c, f \rangle \}$, 该图的一个拓扑序列为 a e b d c f。
14. 一棵二叉树高度为 h , 所有结点的度或为 0, 或为 2, 则这棵二叉树最少有 $2h-1$ 个结点。

二、判断题 (15 分, 每题 1 分, 对写 \checkmark , 错写 \times , 答案需写在括号内, 写在括号外无效。)

1. 线性的数据结构可以顺序存储, 也可以链式存储。非线性的数据结构只能链式存储。 ()
 2. 有向图的邻接表和逆邻接表中的结点数一定相同。 ()
 3. 在采用线性探查法处理冲突的散列表中, 所有同义词在表中相邻。 ()
 4. 完全二叉树肯定是平衡二叉树。 () 平衡二叉树(B树)是一棵空树或它的左右两个子树的高度差的绝对值不超过1, 并且左右两个子树都是一棵平衡二叉树
 5. 一个单链表不能采用折半查找法进行查找。 ()
 6. 若一棵二叉树的任一非叶子结点度为 2, 则该二叉树为满二叉树。 ()
 7. 算法的优劣与算法描述语言无关, 但与所用计算机有关。 ()
 8. 两个栈共享一片连续内存空间时, 为提高内存利用率, 减少溢出机会, 应把两个的栈底分别设在这片内存空间的两端。 ()
 9. 利用拓扑排序, 可检测一个有向图中是否存在环。 ()
 10. 非空满二叉树的结点个数一定为奇数个。 ()
 11. 哈夫曼树的结点总个数一定是奇数。 () 无向图一条边要算两次 (2e) 再加上出发的各个节点 (n) 所以 $2e+n$
 12. 有 e 条边的无向图在其对应的邻接表中有 e 个结点。 ()
 13. 二叉树为二叉排序树的充分必要条件是任一结点的值均大于其左孩子的值、小于其右孩子的值。 ()
 14. 完全二叉树中, 若一个结点没有左孩子, 则它必是叶子结点。 ()
 15. 有一组数据 (15, 9, 6, 8, 20, -1, 7, 4), 用堆排序的筛选方法建立的初始堆为 (-1, 4, 8, 9, 20, 6, 15, 7)。 () https://blog.csdn.net/Senbonzakura_py/article/details/127547232
- X \checkmark X \checkmark \checkmark XX \checkmark \checkmark \checkmark \checkmark XX \checkmark \checkmark

三、综合题：(48 分，每题 6 分)

1. 某二叉树的中序序列为 DBGEACF，后序序列为 DGEBFCA，

(1) 请画出该二叉树；

(2) 将该二叉树转换为相应的森林。

2. 下表展示了在一段文本中每个字母出现的次数。对于这段文本，使用 Huffman 编码比使用等长编码能够节约多少比特的空间？

a	12
e	8
i	15
o	4
u	9

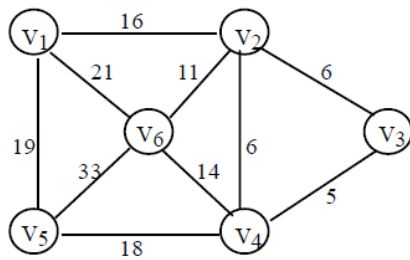
\

3. 已知无向图 G 的邻接矩阵如下，顶点依次为 a、b、c、d、e。

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

- (1) 画出该无向图。
- (2) 根据邻接矩阵，从顶点a出发进行深度优先遍历和广度优先遍历，写出相应的遍历序列。

4. 已知一个无向网如下图所示，要求分别用 Prim 和 Kruskal 算法生成最小代价生成树（假设以 V1 为起点），要求写出构造过程。



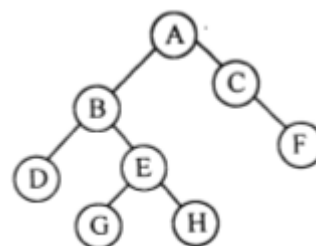
5. 给定一个关键字序列{25, 28, 30, 43, 18, 36, 15, 22}, 请分别写出应用快速排序、简单选择排序、希尔排序 (增量 $d1=5$), 排序第一趟后的结果。

6. 已知二叉树的存储结构为二叉链表, 阅读下面算法:

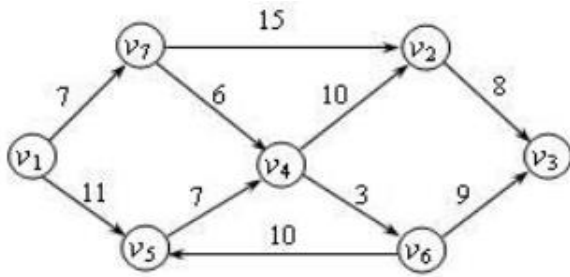
```
typedef struct node{
    DataType data;
    struct node *next;
}ListNode,*LinkedList;
LinkedList Leafhead=NULL;
void inorder(BinTree T)
{ LinkedList s;
  if (T) {
    inorder(T->lchild);
    if ((!T->lchild)&&(!T->rchild)) {
      s=(ListNode*)malloc(sizeof(ListNode));
      s->data=T->data;
      s->next=Leafhead;
      Leafhead=s;
    }
    inorder(T->rchild);
  }
}
```

对于如右图所示的二叉树

- (1).画出执行上述算法后所建立结构;
- (2).说明该算法的功能。



7. 对下图所示的带权有向图，根据 Dijkstra 算法求从源点 V_1 到其它各顶点的最短路径及其长度。



8. 算法填空：

设查找表用顺序表存储，具体结构类型描述如下

typedef struct

```

{   int   key;           //关键字域
    int   otherinfo;     //其他域
} ElemType;

```

typedef struct

```

{   ElemType *elem; //数据元素存储空间基址，建表时按实际长度分配，0 号单元留空
    int length; //表长度
} SSTable;

```

以下为二分查找的非递归算法，在有序表 ST 中查找其关键字等于 key 的数据元素。若找到，则函数值为该元素在表中的位置，否则为 0。

int searchbin(SSTable ST, int key)

```

{   low=1;
    high=ST.length;
    while( _____ )
    {
        mid= _____;
        if(key==ST.elem[mid].key) return mid;
        else if(key<ST.elem[mid].key) high=mid-1;
        else _____;    }
}

```

四、算法设计题（22 分， 第 1、2 题每题 7 分, 第 3 题 8 分）

1. 假设某个单向循环链表的长度大于 1，且表中既无头结点也无头指针。已知 s 指向链表中某个结点的指针，编写算法在链表中删除指针 s 所指结点的前驱结点。

2. 二叉树按二叉链表形式存储，编写下列算法

(1) 建立二叉树；

(2) 求二叉树中度为 2 的结点个数。

3. 已知无向图 G 采用邻接表存储方式，邻接表定义如下，完成下列几个函数。

```
#define MAX_NUM 20    //图的顶点总数
```

```
typedef char VertexType;
```

```
typedef struct ArcNode{    //边结点的类型定义
```

```
    int    adjvex;        // 边的另一顶点的在数组中的位置
```

```
    struct ArcNode    *nextarc; // 指向下一条边的指针
```

```
}ArcNode;
```

```
typedef struct VNode {    //顶点结点和数组的类型定义
```

```
    VertexType    data;    // 顶点信息
```

```
    ArcNode        *firstarc; //指向关联该顶点的边链表
```

```
}VNode, AdjList[MAX_NUM];
```

```
typedef struct{
```

```
    AdjList    vertices;    //顶点数组
```

```
    int        vexnum, arcnum; // 图的当前顶点数和边数
```

```
}ALGraph;    //图定义
```

```
int    LocateVex(ALGraph G,VertexType v)//求顶点 v 在图 G 中的位置
```

```
{//补充程序代码
```

```
}
```

```
int FirstAdjVex(ALGraph G,int v)* v 是图 G 中某个顶点的序号，返回 v 的第 1 个邻接顶点的序号。若顶点在 G 中没有邻接顶点，则返回-1。*/
```

```
{ //补充程序代码
```

```
}
```

系主任 _____

出卷人 _____