

1. 在逻辑上可以把数据结构分成：()。

- A. 动态结构和静态结构 B. 紧凑结构和非紧凑结构
C. 线性结构和非线性结构 D. 内部结构和外部结构

2. L 是线性表，已知 $\text{LengthList}(L)$ 的值是 5，经 $\text{DelList}(L, 2)$ 运算后， $\text{LengthList}(L)$ 的值是 ()。

- A. 2 B. 3 C. 4 D. 5

3. 在数据结构中，与所使用的计算机无关的是 ()。

- A. 物理结构 B. 存储结构 C. 逻辑结构 D. 逻辑和存储结构

4. 下面程序段的时间复杂度为()。

```
for(int i=0;i<m;i++)
    for(int j=0;j<n;j++)
        a[i][j]=i*j;
```

- A. $O(m^2)$ B. $O(n^2)$ C. $O(m*n)$ D. $O(m+n)$

5. 执行下面程序段时，执行 T 语句的次数为()。

```
for(int i=1;i<=n;i++)
    for(int j=1;j<=n;j++)
        T;
```

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

6. 以下算法的时间复杂度为()。

```
void fun(int n)
{
    int i=1;
    while (i<=n)
        i=i*3;
}
```

- A. $O(n)$ B. $O(n^2)$ C. $O(\log_2 n)$ D. $O(\log_3 n)$

6. 下列程序段的时间复杂度是()。

```
□ i=1;
while(i<=n)
□ i=i*2
```

A、 $O(\log 2n)$

B、 $O(n)$

C、 $O(n \log 2n)$

D、 $O(n^2)$

7. 在一个链队中, 假设 f 和 r 分别为队首和队尾指针, 删除一个结点的运算是()。

A、 $r = f \rightarrow \text{next}$

B、 $r = r \rightarrow \text{next}$

C、 $f = f \rightarrow \text{next}$

D、 $f = r \rightarrow \text{next}$

8. 设有编号为 1, 2, 3, 4 的四辆列车, 顺序进入一个栈结构的站台, 下列不可能的出站顺序为()

A. 1234

B. 1243

C. 1324

D. 1423

9. 某算法的时间代价为 $T(n) = 300n + 20n \log_2 n + 10n^2$, 其时间复杂度为()。

A、 $O(n)$

B、 $O(n \log_2 n)$

C、 $O(n^2)$

D、 $O(1)$

10. 用二分查找方法查找长度为 n 的线性表时, 每个元素的平均查找长度为()。

A、 $O(n)$

B、 $O(\log 2n)$

C、 $O(n^2)$

D、 $O(1)$

11. 一组记录的排序码为 (25, 48, 16, 35, 79, 82, 23, 40), 其中含有 4 个长度为 2 的有序表, 按归并排序的方法对该序列进行一趟归并后的结果为: ()。

A, 16 25 35 48 23 40 79 82

B. 16 25 35 48 79 82 23 36 40 72

C. 16 25 48 35 79 82 23 36 40 72

D. 16 25 35 48 79 23 36 40 72 82

12. 二分查找有序表 {4, 6, 10, 12, 20, 30, 50, 70, 88, 100}, 若查找表中元素 58, 则它将依次与表中()比较大小, 查找结果是失败。

A. 30, 88, 70, 50

B. 20, 70, 30, 50

C. 20, 50

D. 30, 88, 50

13. 若串 $S = \text{"software"}$, 其真子串的个数是()。

A. 8

B. 37

C. 36

D. 9

14. 二分查找有序表 {4, 6, 10, 12, 20, 30, 50, 70, 88, 100}, 若查找表中元素 58, 则它将依次与表中()比较大小, 查找结果是失败。

A. 30, 88, 70, 50

B. 20, 70, 30, 50

C. 20, 50

D. 30, 88, 50

15. 用链表表示线性表的优点是()。

A、便于随机存取

B、花费的存储空间较顺序存储少

C、便于插入和删除

D、数据元素的物理顺序与逻辑顺序相同

16. 以下论述正确的是()。

A. 空串与空格串是相同的

B. "tel"是"Teleptone"的子串

C. 空串是零个字符的串

D. 空串的长度等于 1

17. 对于一个具有 N 个顶点的图, 如果我们采用邻接矩阵法表示, 则此矩阵的维数应该是()。

A、 $(N-1) \times (N-1)$

B、 $N \times N$

C、 $(N+1) \times (N+1)$

D、不确定

18. 在有 n 个叶子结点的哈夫曼树中，其结点总数为（ ）。

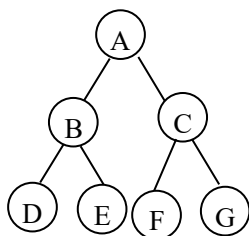
- A、 不确定 B、 $2n$ C、 $2n+1$ D、 $2n-1$

19. 以下序列不是堆的是（ ）

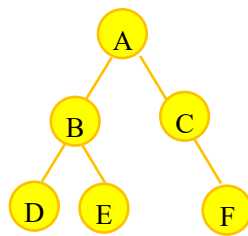
- A、 {100, 85, 98, 77, 80, 60, 82, 40, 20, 10, 66}
B、 {100, 98, 85, 82, 80, 77, 66, 60, 40, 20, 10}
C、 {10, 20, 40, 60, 66, 77, 80, 82, 85, 98, 100}
D、 {100, 85, 40, 77, 80, 60, 66, 98, 82, 10, 20}

20. 下列 4 棵树中，（ ）不是完全二叉树。

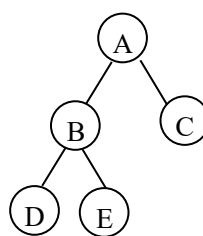
A.



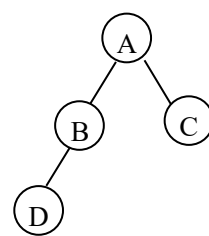
B.



C.



D.



21. 下面关于图的存储结构的叙述中正确的是（ ）。

- A、 用邻接矩阵存储图，占用空间大小只与图中顶点数有关，而与边数无关
B、 用邻接矩阵存储图，占用空间大小只与图中边数有关，而与顶点数无关
C、 用邻接表存储图，占用空间大小只与图中顶点数有关，而与边数无关
D、 用邻接表存储图，占用空间大小只与图中边数有关，而与顶点数无关

22. 设有一个顺序栈 S ，元素 A, B, C, D, E, F ，依次进栈，如果六个元素出栈的顺序是 B, D, C, F, E, A ，则栈的容量至少应是（ ）。

- A. 3 B. 4 C. 5 D. 6

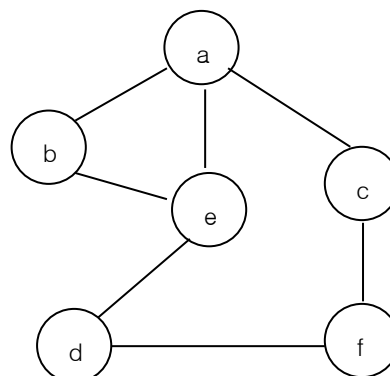
23. 如下图所示，从顶点 a 出发，按广度优先进行遍历，则可能得到的一种顶点序列为（ ）。

A. a, b, e, c, d, f

B. a, b, e, c, f, d

C. a, e, b, c, f, d

D. a, e, d, f, c, b



24. 对稀疏矩阵进行压缩存储是为了（ ）。

- A. 降低运算时间 B. 节约存储空间
- C. 便于矩阵运算 D. 便于输入和输出
25. 用 5 个权值 {3, 2, 4, 5, 1} 构造的哈夫曼树的带权路径长度是 ()。
- A. 32 B. 33 C. 34 D. 15
26. 设一组初始记录关键字序列为 (50, 40, 95, 20, 15, 70, 60, 45)，则以增量 $d=4$ 的一趟希尔排序结束后前 4 条记录关键字为 ()。
- A. 40, 50, 20, 95 B. 15, 40, 60, 20
- C. 15, 20, 40, 45 D. 45, 40, 15, 20
27. 设一组初始记录关键字序列为 (45, 80, 55, 40, 42, 85)，则以第一个记录关键字 45 为基准而得到一趟快速排序的结果是 ()。
- A. 40, 42, 45, 55, 80, 83 B. 42, 40, 45, 80, 85, 88
- C. 42, 40, 45, 55, 80, 85 D. 42, 40, 45, 85, 55, 80
28. 设一组初始记录关键字序列为 (25, 50, 15, 35, 80, 85, 20, 40, 36, 70)，其中含有 5 个长度为 2 的有序子表，则用归并排序的方法对该记录关键字序列进行一趟归并后的结果为 ()。
- A. 15, 25, 35, 50, 20, 40, 80, 85, 36, 70
- B. 15, 25, 35, 50, 80, 20, 85, 40, 70, 36
- C. 15, 25, 35, 50, 80, 85, 20, 36, 40, 70
- D. 15, 25, 35, 50, 80, 20, 36, 40, 70, 85
29. 最短路径的生成算法可用 ()。
- A. 普里姆算法 B. 克鲁斯卡尔算法 C. 迪杰斯特拉算法 D. 哈夫曼算法
- 30 无向图的邻接矩阵是一个 ()。
- A. 对称矩阵 B. 零矩阵 C. 上三角矩阵 D. 对角矩阵
31. 一个有序顺序表有 255 个对象，采用顺序搜索法查表，成功平均搜索长度为 ()。
- A. 128 B. 127 C. 126 D. 255
32. 设输入序列是 1、2、3、……、n，经过栈的作用后输出序列的第一个元素是 n，则输出序列中第 i 个输出元素是 ()。
- A. $n-i$ B. $n-1-i$ C. $n+1-i$ D. 不能确定
33. 对顺序存储的线性表，设其长度为 n，在任何位置上插入或删除操作都是等概率的。插入一个元素时平均要移动表中的 () 个元素。

A、 $n/2$

B、 $(n+1)/2$

C、 $(n-1)/2$

D、 n

34. 设环形队列中数组的下标为 $0 \sim N-1$ ，其队头、队尾指针分别为 $front$ 和 $rear$ ($front$ 指向队列中队头元素的前一个位置， $rear$ 指向队尾元素的位置)，则其元素个数为 ()。

A. $rear-front$

B. $rear-front-1$

C. $(rear-front) \% N + 1$

D. $(rear-front+N) \% N$

35. 经过下列栈的运算后， $SEmpty(s)$ 的值是 ()。

InitStack(s) (初始化栈); Push(s,a); Push(s,b); Pop(s,x); Pop(s,x);

A. a

B. b

C. 1

D. 0

36. 中缀表达式 $a*(b+c)-d$ 的对应的后缀表达式是 ()。

A. $a b c d * + -$

B. $a b c + * d -$

C. $a b c * + d -$

D. $- + * a b c d$

37. 中缀表达式“ $2*(3+4)-1$ ”的后缀表达式是 ()，其中 # 表示一个数值的结束。

A. $2\#3\#4\#1\#+-$

B. $2\#3\#4\#+*1\#-$

C. $2\#3\#4\#+1\#-$

D. $-+*2\#3\#4\#1\#$

38. 用直接插入排序法对下面的四个序列进行由小到大的排序，元素比较次数最少的是 ()。

A. 94, 32, 40, 90, 80, 46, 21, 69

B. 21, 32, 46, 40, 80, 69, 90, 94

C. 32, 40, 21, 46, 69, 94, 90, 80

D. 90, 69, 80, 46, 21, 32, 94, 40

39. 堆的形状是一棵_____。

A. 满二叉树

B. 二叉判定树

C. 平衡二叉树

D. 完全二叉树

40. 对于一个具有 n 个顶点的有向图的边数最多有 ()。

A. n

B. $n(n-1)$

C. $n(n-1)/2$

D. $2n$

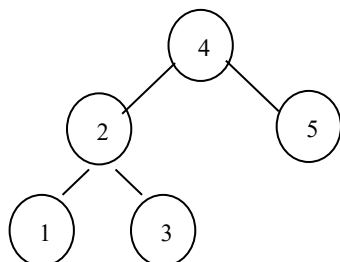
41. 不可能生成下图二叉排序树的关键字的序列是 ()。

A. 4 5 3 1 2

B. 4 2 5 3 1

C. 4 5 2 1 3

D. 4 2 3 1 5



42. 函数 substr(“DATASTRUCTURE”, 5, 9)的返回值为 ()。

- A、 “STRUCTURE” B、 “DATA”
C、 “ASTRUCTUR” D、 “DATASTRUCTURE”

43. 有 64 个结点的完全二叉树的深度为 () (根的层次为 1)。

- A、8 B、7 C、6 D、5

44. 具有 n ($n > 1$) 个结点的完全二叉树中, 结点 i ($2i > n$) 的左孩子结点是 ()。

- A. $2i$ B. $2i+1$ C. $2i-1$ D. 不存在

45. 将一棵有 100 个结点的完全二叉树从上到下, 从左到右依次对结点编号, 根结点的编号为 1, 则编号为 45 的结点的左孩子编号为 ()。

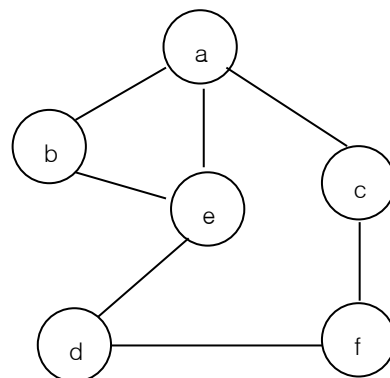
- A. 46 B. 47 C. 90 D. 91

46. 设有序表中有 1000 个元素, 则用二分查找查找元素 x 最多需要比较 () 次。

- A、 25 B、 10 C、 7 D、 1

47. 如下图所示, 从顶点 a 出发, 按深度优先进行遍历, 则可能得到的一种顶点序列为 ()。

- A. a, b, e, c, d, f
B. a, c, f, e, b, d
C. a, e, b, c, f, d
D. a, e, d, f, c, b



48. 在一个有向图中, 所有顶点的入度之和等于所有顶点的出度之和的 () 倍。

- A. $1/2$ B. 1 C. 2 D. 4

49. 在一棵具有五层的满二叉树中, 结点的总数为 ()

- A. 16 B. 31 C. 32 D. 33

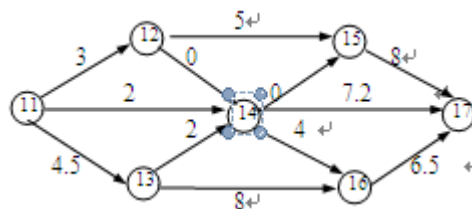
50. 有拓扑排序的图一定是 ()。

- A、有环图 B、无向图 C、强连通图 D、有向无环图

51. 下列 () 不是利用查找表中数据元素的关系进行查找的方法。

- A. 平衡二叉树 B. 有序表的查找
C. 散列查找 D. 二叉排序树的查找

1. 数据结构通常有下列 4 类基本结构：集合、(线性结构)、树型结构、图型结构
2. 数据的基本单位是(数据元素)，最小单位是(数据项)。
3. 两个串是相等的，当且仅当两个串的长度相等且(各对应位置)的字符都相同。
4. 一棵度为 3 的树中，有 3 度结点 100 个，有 2 度结点 200 个，那么叶子结点的个数 401。
5. 在具有 n 个存储单元的队列中，队满时队中共有(n)个元素。
6. 若一棵二叉树有 1001 个结点，且度数为 1 的结点数为 0，则叶子结点的个数 501。
7. 已知元素入栈先后为 ABCDE，若 C 为第一个出栈元素，则下一个出栈的元素可能 B、D、E。
8. 设有向图 G 中有 n 个顶点 e 条有向边，所有的顶点入度数之和为 d，则 e 和 d 的关系为($e=d$)。
9. 假设有一个顺序栈 A，其中元素 a1, a2, a3, a4, a5, a6 依次进栈，如果已知六个元素出栈的顺序是 a2, a3, a4, a6, a5, a1，则此栈容量至少应该为 3。
10. 有 20 个结点的完全二叉树，编号为 10 的结点的父结点的编号是 5。
11. 一个连通图的生成树是一个极大连通子图，n 个顶点的生成树有 n-1 条边。
12. 设循环队列的容量为 40 (序号从 0 到 39)，现经过一系列的入队和出队运算后，有 front=11, rear=19，则循环队列中还有 8 个元素。
13. 在如下图所示的网络计划图中关键路径是 (11, 13, 16, 17)，全部计划完成的时间是 19。



14. 将一个 N 阶矩阵 A 的上三角部分按行优先压缩存放于一个一维数组 B 中，A[0][0] 存放于 B[0] 中，则 A[i][j] 在 $i \leq j$ 时将存放于数组 B 的 ($i(2n-i+1)/2+j-i$) 位置
15. 各结点左右子树深度之差的绝对值至多为 1 的二叉树称谓平衡二叉树。
16. 对于长度为 n 的线性表，若采用二分查找，则时间复杂度为： $O(\log_2^n)$ 。
17. 每次使两个相邻有序表合并成一个有序的排序方法叫做 (二路归并) 排序。
18. 求子串函数 SubStr("Today is 30 July, 2005", 13, 4) 的结果是： July。
19. 设 S="My Music"，则 LenStr(s)= 8。
1. 已知模式串 pat=' abcaabaa'，写出该模式串的 next 函数值和 nextval 值。

下标	0	1	2	3	4	5	6
关键字	a	b	c	a	b	a	a
next	-1	0	0	0	1	2	1
nextval	-1	0	0	-1	0	2	1

2、已知模式串 pat=“abaabc”，写出该模式串的 next 函数值和 nextval 值；

下标	0	1	2	3	4	5
关键字	a	b	a	a	b	c
next	-1	0	0	1	1	2
nextval	-1	0	-1	1	0	2

3、给出字符串‘abacabaaad’在 KMP 算法中的 next 和 nextval 数组。

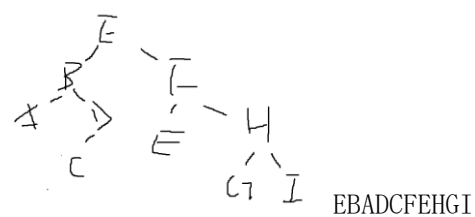
下标	0	1	2	3	4	5	6	7	8	9
关键字	a	b	a	c	a	b	a	a	a	d
next	-1	0	0	1	0	1	2	3	1	1
nextval	-1	0	-1	1	-1	0	-1	3	1	1

4. 已知模式串 pat=' abaabaab' , 写出该模式串的 next 函数值和 nextval 值；

下标	0	1	2	3	4	5	6	7
关键字	a	b	a	a	b	a	a	b
next	-1	0	0	1	1	2	3	4
nextval	-1	0	-1	1	0	-1	1	0

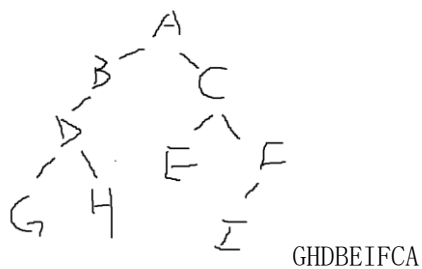
5.已知一棵二叉树的后序遍历和中序遍历的序列分别为：ACDBGIHFE 和 ABCDEFGHI。

请画出该二叉树，并写出它的前序遍历的序列。



6. 已知一棵二叉树的前序遍历和中序遍历的序列分别为：ABDGHCEFI 和 GDHBAECIF。

请画出此二叉树，并写出它的后序遍历的序列。



7. 假定一棵二叉树的括号表示为 $A(B(, D(G)), C(E, F))$ ，请分别写出对它的先序、中序、层次的遍历序列。

ABDGCEF BGDAECF GDBEFCA

8. 已知某二叉树的先序遍历次序为：abcdefg，中序遍历次序为：badfegc，画出该二叉树，并在该二叉树上建立中序线索。



9. 已知一棵度为 3 的树中，有度数为 3 的结点 100 个，度数为 2 的结点 200 个，求叶子结点的个数，并给出推导过程。

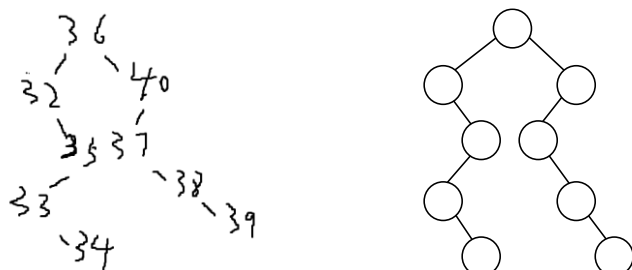
$$n = n_0 + n_1 + n_2 + n_3$$

$$n = 0 \cdot n_0 + 1 \cdot n_1 + 2 \cdot n_2 + 3 \cdot n_3 + 1$$

$$n_0 + n_2 + n_3 = 2 \cdot n_2 + 3 \cdot n_3 + 1$$

$$n_0 = 401$$

10. 一棵二叉排序树的结构如图 1 所示，其中各结点的关键字依次为 32~40，请标出各结点的关键字。



11. 数组元素 $a[0..2][0..3]$ 的首地址为 2000，元素长度是 4，求 $LOC[1,2]$ 。

$$2000 + 6 \times 4 = 2024$$

12. 广义表 $LS = (a, (b), ((c, (d))))$ 求：表头，表尾，长度，深度。

a $((b), ((c, (d))))$ 3 4

13. 已知二维数组 $A[6][10]$ ，每个数组元素占 4 个存储单元，若按行优先顺序存放数组元素 $a[3][5]$ 的存储地址是 1000，求 $a[0][0]$ 的存储地址？（写出公式计算过程）。

$$a[3][5] \text{ 前共有元素个数} = 3 \times 10 + 5 = 35$$

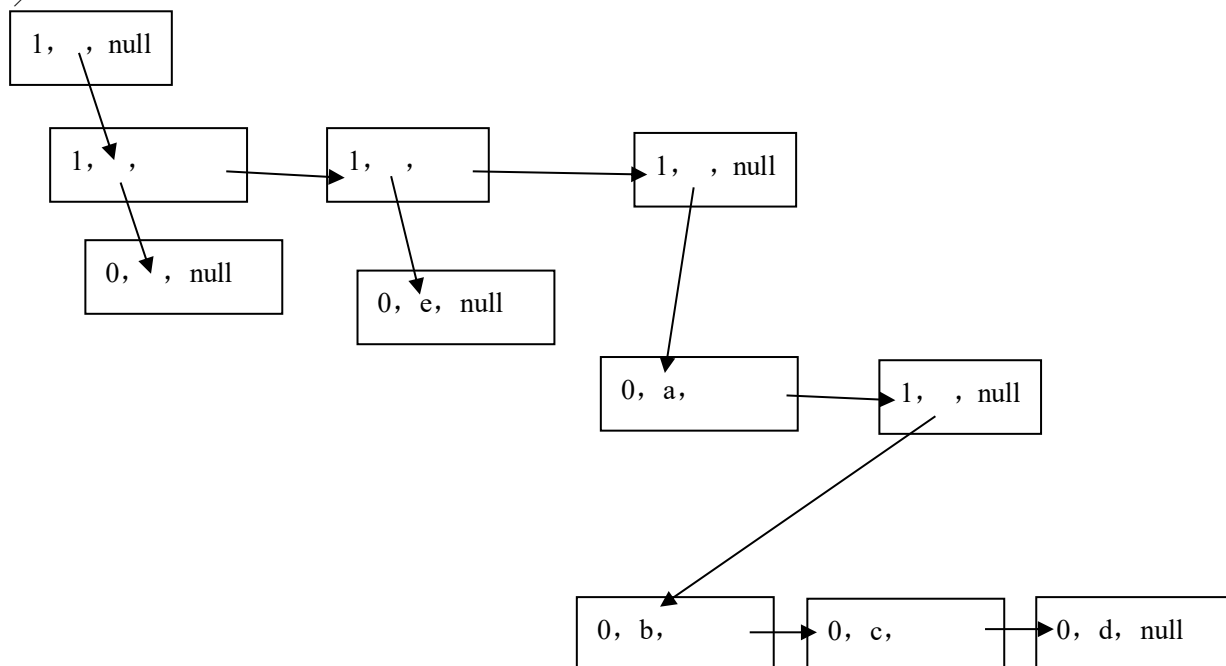
$$a[0][0] \text{ 的存储地址} = 1000 - 35 \times 4 = 860$$

14 已知广义表 $LS = ((a, b), c, (d, (e, f), g))$ 求：表头，表尾，长度，深度。

(a, b) $(c, (d, (e, f), g))$ 3 3

15、画出广义表 $LS = ((), (e), (a, (b, c, d)))$ 的头尾链表存储结构。

LS->



1. 设单链表的结点的结构为 $ListNode = (data, link)$ ，阅读下面的函数，指出它所实现的功能是什么。

```

Int unknown(ListNode *Ha)
{
    Int n=0;
    ListNode *p=Ha->link;
    While(p)
    {

```

```

        n++;

        p=p->next;

    }

    Return (n);

}

```

计算单链表长度

2. 对于顺序表 L ，指出以下算法的功能。

```

void fun(SqList *&L)

{ int i,j=0;

    for (i=1;i<L->length;i++)

        if (L->data[i]>L->data[j])

            j=i;

    for (i=j;i<L->length-1;i++)

        L->data[i]=L->data[i+1];

    L->length--;

}

```

找最大值位置并且把最大值删除

3. 写结果

```

main()

{Stack S;

char x,y;

S.InitStack();

X=' c' ;y=' k' ;

S.Push(x);S.Push( 'a' );S.Push(y);

S.Pop(S,x);S.Push( 't' );S.Push( 's' );

while(!S.IsEmpty())

{ S.Pop(S,x);

printf( "%c" ,x);}

printf( "%c" ,y);

}
stack

```

4 假设二叉树采用二叉链存储结构存储，设计一个递归算法，计算一棵给定二叉树的所有结点数。

```
int NodesCount(BTNode *b)
{
    if(b==NULL)
        return 0;
    else
        return NodesCount(b->lchild)+NodesCount(b->rchild)+1;
}
```

5 求二叉树高度的算法。（递归）

```
int BTHHeight(BTNode *b)
{
    int lchild,rchild;
    if(b==NULL) return(0);
    else{
        lchild= BTHHeight(b->lchild);
        rchild= BTHHeight(b->rchild);
        return (lchild>rchild)?(lchild+1):(rchild+1);
    }
}
```

6、编写在一有序顺序表中插入数据元素 X 的算法 INSERT(L, X)

```
void INSERT(SqList &L,ElemType X){
    int i=0;
    while(x<L.data[i]&&i<=L.length-1)
        ++i;
    for(int j=L.length-1;j>=i;j--)
        L.data[j+1]=data[j];
    L.data[i]=x;
    L.length++;
}
```

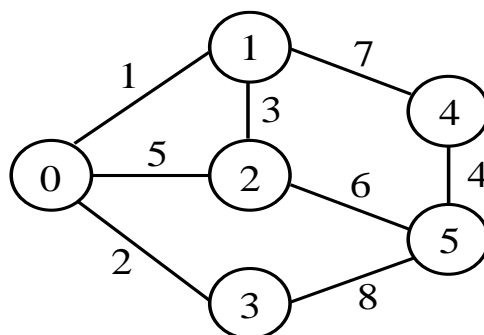
7. 假设二叉树采用二叉链存储结构存储，设计一个递归算法，输出给定二叉树的所有叶子结点。

```
void DispLeaf(BTNode *b) {
    if(b!=NULL) {
        if(b->lchild==NULL&&b->rchild==NULL)
            printf( "%c" ,b->data);
        DispLeaf(b->lchild);
        DispLeaf(b->rchild);
    }
}
```

1. 假设用于通信的电文仅由 A、B、C、D、E、F、G 8 个字母组成，字母在电文中出现的频率分别为 7，19，2，6，32，3，21，10。试为这 8 个字母设计哈夫曼树及哈夫曼编码。

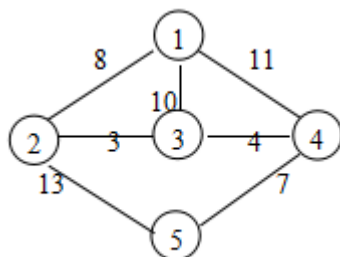
A:1010,B:00,C:10000,D:1001,E:11,F:10001,G:01,H:1011

2、对于下图所示的带权无向图，给出利用普里姆算法（从顶点 0 开始构造）和克鲁斯卡尔算法构造出的最小生成树，并按求解的顺序给出最小生成树的所有边，每条边用 (i, j) 表示）。



$\{(0, 1), (0, 3), (1, 2), (2, 5), (5, 4)\}$
 $\{(0, 1), (0, 3), (1, 2), (4, 5), (2, 5)\}$

3、网 G 的图形如下，试写出邻接矩阵，并画出它的一棵最小生成树。



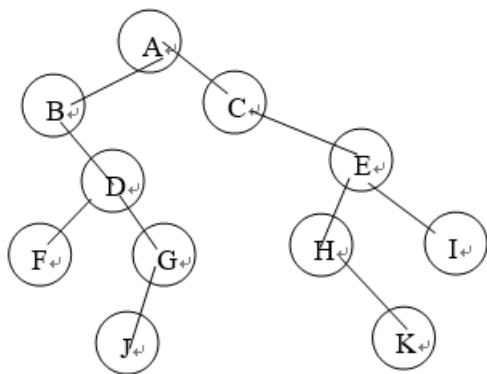
邻接矩阵：

$$A = \begin{pmatrix} 0 & 8 & 10 & 11 & \infty \\ 8 & 0 & 3 & \infty & 13 \\ 10 & 3 & 0 & 4 & \infty \\ 11 & \infty & 4 & 0 & 7 \\ \infty & 13 & \infty & 7 & 0 \end{pmatrix}$$

$\{(2, 3), (3, 4), (4, 5), (2, 1)\}$

4、已知某二叉树的中序和后序遍历序列分别为 BFDJGACHKEI 和 FJGDBKHIECA，

(1) 请画出该二叉树。



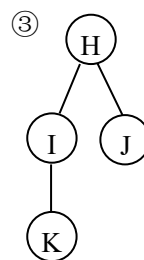
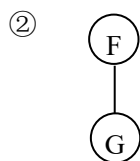
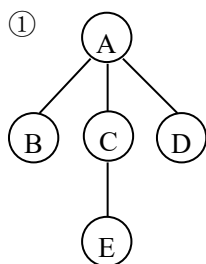
(2) 树的深度是多少？

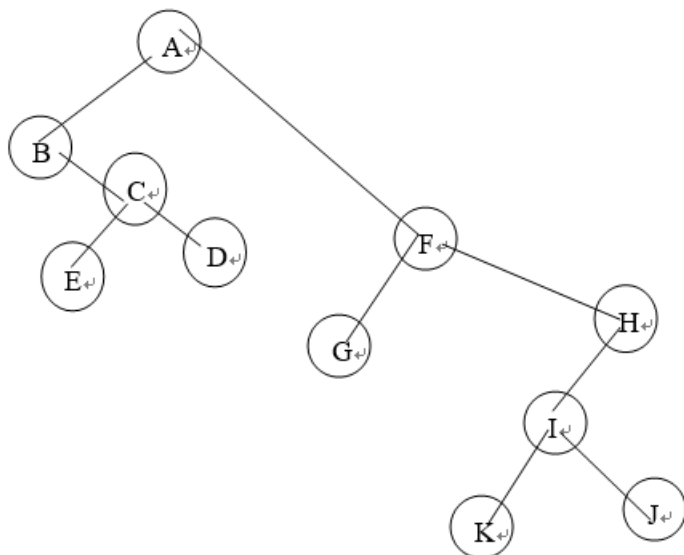
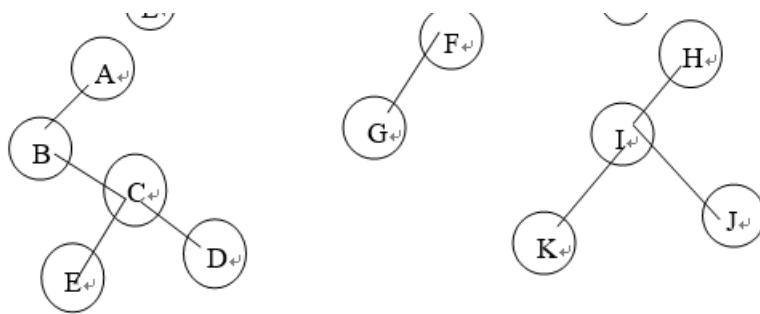
5

(3) 给出该二叉树的先序遍历序列。

ABDFGJCEHKEI

5、把下列森林转换为二叉树（写出过程）

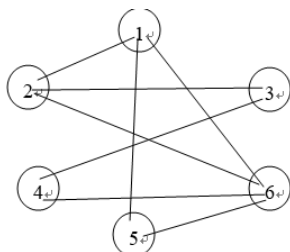




6. 带权无向图G（顶点分别为V 1，V 2，V 3，V 4，V 5，V 6）的邻接矩阵是A

$$A = \begin{matrix} & \begin{matrix} v1 & v2 & v3 & v4 & v5 & v6 \end{matrix} \\ \begin{matrix} v1 \\ v2 \\ v3 \\ v4 \\ v5 \\ v6 \end{matrix} & \begin{pmatrix} 0 & 6 & \infty & \infty & 6 & 3 \\ 6 & 0 & 5 & \infty & \infty & 4 \\ \infty & 5 & 0 & 6 & \infty & \infty \\ \infty & \infty & 6 & 0 & \infty & 7 \\ 6 & \infty & \infty & \infty & 0 & 2 \\ 3 & 4 & \infty & 7 & 2 & 0 \end{pmatrix} \end{pmatrix}$$

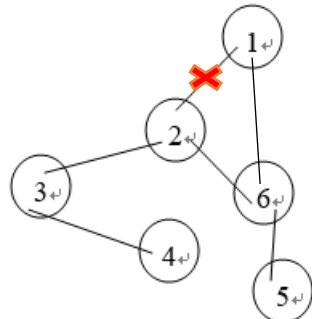
要求：（1）画出图G。（3分）



(2) 分别写出从V 1 出发的深度优先遍历序列和广度优先遍历序列。(4 分)

深度:1, 2, 3, 4, 6, 5 广度:1, 2, 5, 6, 3, 4

(3) 画出一棵最小生成树 (3 分)



7、设哈希表的长度 $m=13$ ；哈希函数为 $H(K)=K\%m$ ，给定的关键码序列为 19，14，23，01，68，20，84，27，55，11，试填出用线性探查法和链地址法解决冲突时所构造的哈希表。

并求在每种哈希表上成功查找的 ASL。

线性探测法：

下标	0	1	2	3	4	5	6	7	8	9	10	11	12
关键字		14	01	68	27	55	19	20	84		23	11	
探测次数		1	2	1	4	3	1	1	3		1	1	

ASL 成功 $= (1+2+1+4+3+1+1+3+1+1)/10=18/10=9/5$

链地址法：

0

1→27→01→14

2

3→55→68

4

5

6→84→19

7→20

8

9

10→23

11→11

12

ASL 成功 = $(1 \times 6 + 2 \times 3 + 3) / 10 = 15 / 10 = 3 / 2$

8、已知关键字序列为 (34, 23, 56, 32, 45, 58, 89, 20, 25, 50)，分别用下列排序方法进行排序，

分别写出每趟排序结果，并指出算法的稳定性。1) 快速排序 2) 希尔排序 3) 堆排序

(1)

(25, 23, 20, 32, 34, 58, 89, 45, 56, 50)

(20, 23, 25, 32, 34, 50, 56, 45, 58, 89)

(20, 23, 25, 32, 34, 45, 50, 56, 58, 89) 划分的子表为 1 或 0 停止即可

不稳定

(2)

(34, 23, 20, 25, 45, 58, 89, 56, 32, 50)

(20, 23, 32, 25, 34, 50, 45, 56, 89, 58)

(20, 23, 25, 32, 34, 45, 50, 56, 58, 89)

不稳定

(3)

(58, 50, 56, 32, 45, 34, 23, 20, 25, 89)

(56, 50, 34, 32, 45, 25, 23, 20, 58, 89)

(50, 45, 34, 32, 20, 25, 23, 56, 58, 89)

(45, 32, 34, 23, 20, 25, 50, 56, 58, 89)

(34, 32, 25, 23, 20, 45, 50, 56, 58, 89)

(32, 23, 25, 20, 34, 45, 50, 56, 58, 89)

(25, 23, 20, 32, 34, 45, 50, 56, 58, 89)

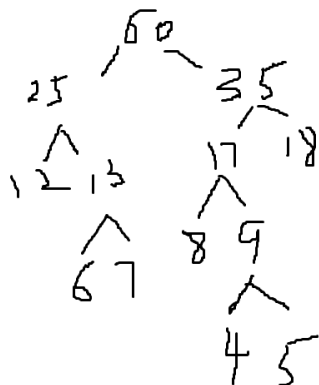
(23, 20, 25, 32, 34, 45, 50, 56, 58, 89)

(20, 23, 25, 32, 34, 45, 50, 56, 58, 89)

不稳定

9. 给定一个权集 $W = \{4, 5, 7, 8, 6, 12, 18\}$ 。

(1) 画出哈夫曼树 (5 分)



(2) 设计它们的哈夫曼编码 (3 分)

4:1010 5:1011 7:011 8:100 6:010 12:00 18:11

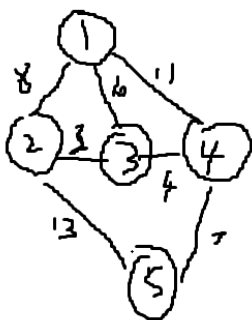
(3) 计算 WPL 值。(2 分)

$$WPL=4*(4+5)+3*8+2*(18+12)+3*(6+7)=159$$

10. 网 G 的邻接矩阵如下, 试画出该图, 并画出它的一棵最小生成树。

$$\begin{bmatrix} 0 & 8 & 10 & 11 & 0 \\ 8 & 0 & 3 & 0 & 13 \\ 10 & 3 & 0 & 4 & 0 \\ 11 & 0 & 4 & 0 & 7 \\ 0 & 13 & 0 & 7 & 0 \end{bmatrix}$$

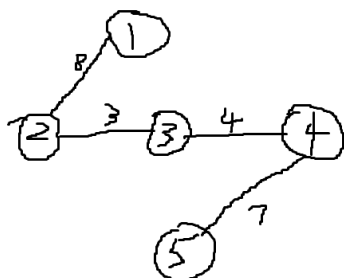
要求: (1) 画出图 G。(3 分)



(2) 分别写出从 V1 出发的深度优先遍历序列和广度优先遍历序列。(4 分)

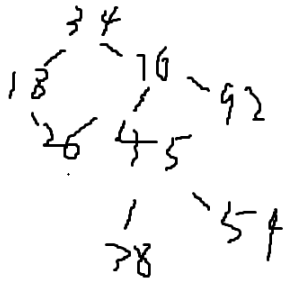
深度: V1,V2,V3,V4,V5 广度: V1,V2,V3,V4,V5

(3) 画出一棵最小生成树 (3 分)



11. 对于给定结点的关键字集合 $K=\{34, 76, 45, 18, 26, 54, 92, 38\}$,

(1) 试构造一棵二叉排序树;



(2) 查找 54 需要比较几次? 查找 100 的比较次数?

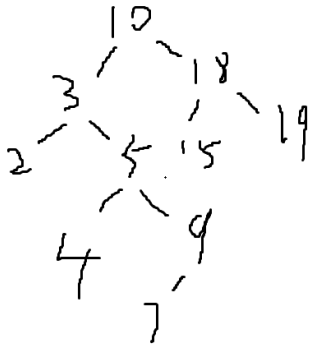
4 3

(3) 求等概率情况下查找成功的平均查找长度 ASL。

$$\text{ASL 成功} = (1 + 2 \times 2 + 3 \times 3 + 4 \times 2) / 8 = 22 / 8 = 11 / 4$$

12. 对于给定结点的关键字集合 $K=\{10, 18, 3, 5, 19, 2, 4, 9, 7, 15\}$,

(1) 试构造一棵二叉排序树;



(2) 求等概率情况下的平均查找长度 ASL。

$$\text{ASL 成功} = (1 + 2 \times 2 + 3 \times 4 + 4 \times 2 + 5) / 10 = 3$$

$$\text{ASL 失败} = (3 \times 6 + 4 \times 3 + 5 \times 2) / 11 = 40 / 11$$

(3) 查找 4 需要比较几次? 查找 8 的比较次数?

4 5

13. 已知无向图的顶点集合为 $\{A, B, C, D, E, F, G\}$, 边集合为 $\{(A, B, 5), (B, C, 6),$

$(A, E, 8), (B, D, 4), (C, F, 5), (D, E, 3), (D, F, 9), (F, G, 7), (E, G, 3), (D, G, 3)\}$, 其中边集合中的数字信息为边上的权值, 画出该无向图的邻接表和最小生成树, 并以邻接表为基础分别写出深度、广度优先遍历序列 (要求从结点 A 开始)。

邻接表:

A→B→E

B→A→C→D

C→B→F

D→B→E→F→G

E→A→D→G

F→C→D→G

G→D→E→F

最小生成树:

{(D, E), (E, G), (B, D), (A, B), (C, F), (B, C)}

深度: ABCFDEG 广度: ABECDFG

14、已知某有向图如图所示:

(1) 给出其邻接表存储结构

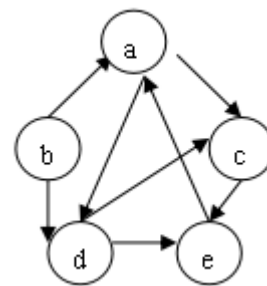
a→c→d

b→a→d

c→e

d→c→e

e→a



(2) 给出其深度优先遍历次序。

baced

(3) 给出其广度优先遍历次序。

badce

(4) 给出各强连通分量。

