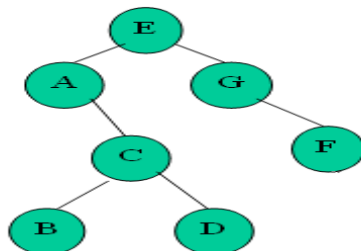


模拟试卷一

一、 单选题（每题 2 分，共 20 分）

- 以下数据结构中哪一个是线性结构？（ ）
A. 有向图 B. 队列 C. 线索二叉树 D. B 树
- 在一个单链表 HL 中，若要在当前由指针 p 指向的结点后面插入一个由 q 指向的结点，则执行如下（ ）语句序列。
A. $p=q$; $p \rightarrow next=q$; B. $p \rightarrow next=q$; $q \rightarrow next=p$;
C. $p \rightarrow next=q \rightarrow next$; $p=q$; D. $q \rightarrow next=p \rightarrow next$; $p \rightarrow next=q$;
- 以下哪一个不是队列的基本运算？（ ）
A. 在队列第 i 个元素之后插入一个元素 B. 从队头删除一个元素
C. 判断一个队列是否为空 D. 读取队头元素的值
- 字符 A、B、C 依次进入一个栈，按出栈的先后顺序组成不同的字符串，至多可以组成（ ）个不同的字符串？
A. 14 B. 5 C. 6 D. 8
- 由权值分别为 3,8,6,2 的叶子生成一棵哈夫曼树，它的带权路径长度为（ ）。
A. 11 B. 35 C. 19 D. 53



图一

- 该二叉树结点的前序遍历的序列为（ ）。
A. E、G、F、A、C、D、B B. E、A、G、C、F、B、D
C. E、A、C、B、D、G、F D. E、G、A、C、D、F、B
- 该二叉树结点的中序遍历的序列为（ ）。
A. A、B、C、D、E、G、F B. E、A、G、C、F、B、D
C. E、A、C、B、D、G、F D. B、D、C、A、F、G、E
- 该二叉树的按层遍历的序列为（ ）。
A. E、G、F、A、C、D、B B. E、A、C、B、D、G、F
C. E、A、G、C、F、B、D D. E、G、A、C、D、F、B
- 下面关于图的存储的叙述中正确的是（ ）。
A. 用邻接表法存储图，占用的存储空间大小只与图中边数有关，而与结点个数无关
B. 用邻接表法存储图，占用的存储空间大小与图中边数和结点个数都有关
C. 用邻接矩阵法存储图，占用的存储空间大小与图中结点个数和边数都有关
D. 用邻接矩阵法存储图，占用的存储空间大小只与图中边数有关，而与结点个数无关
- 设有关键码序列(q, g, m, z, a, n, p, x, h)，下面哪一个序列是从上述序列出发建堆的结果？（ ）
A. a, g, h, m, n, p, q, x, z B. a, g, m, h, q, n, p, x, z
C. g, m, q, a, n, p, x, h, z D. h, g, m, p, a, n, q, x, z

二、 填空题（每空 1 分，共 26 分）

- 数据的物理结构被分为_____、_____、_____和_____四种。
- 对于一个长度为 n 的顺序存储的线性表，在表头插入元素的时间复杂度为_____，在表尾插入元素的时间复杂度为_____。
- 向一个由 HS 指向的链栈中插入一个结点时 p 时，需要执行的操作是_____；

删除一个结点时，需要执行的操作是_____（假设栈不空而且无需回收被删除结点）。

4. 对于一棵具有 n 个结点的二叉树，一个结点的编号为 $i (1 \leq i \leq n)$ ，若它有左孩子则左孩子结点的编号为_____，若它有右孩子，则右孩子结点的编号为_____，若它有双亲，则双亲结点的编号为_____。
5. 当向一个大根堆插入一个具有最大值的元素时，需要逐层_____调整，直到被调整到_____位置为止。
6. 以二分查找方法从长度为 10 的有序表中查找一个元素时，平均查找长度为_____。
7. 表示图的三种常用的存储结构为_____、_____和_____。
8. 对于线性表 (70, 34, 55, 23, 65, 41, 20) 进行散列存储时，若选用 $H(K) = K \% 7$ 作为散列函数，则散列地址为 0 的元素有_____个，散列地址为 6 的有_____个。
9. 在归并排序中，进行每趟归并的时间复杂度为_____，整个排序过程的时间复杂度为_____，空间复杂度为_____。
10. 在一棵 m 阶 B _树上，每个非树根结点的关键字数目最少为_____个，最多为_____个，其子树数目最少为_____，最多为_____。

三、运算题（每题 6 分，共 24 分）

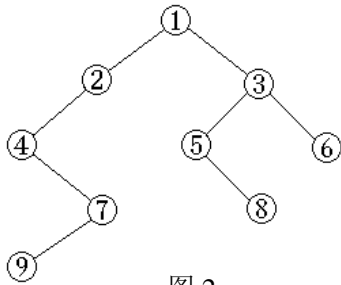


图 2

1. 写出下列中缀表达式的后缀形式：
 - (1) $3X/(Y-2)+1$
 - (2) $2+X*(Y+3)$
2. 试对图 2 中的二叉树画出其：
 - (1) 顺序存储表示的示意图；
 - (2) 二叉链表存储表示的示意图。
3. 判断以下序列是否是小根堆？如果不是，将它调整为小根堆。
 - (1) { 12, 70, 33, 65, 24, 56, 48, 92, 86, 33 }
 - (2) { 05, 23, 20, 28, 40, 38, 29, 61, 35, 76, 47, 100 }
4. 已知一个图的顶点集 V 和边集 E 分别为：

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

$$E = \{(1, 2)3, (1, 3)5, (1, 4)8, (2, 5)10, (2, 3)6, (3, 4)15, (3, 5)12, (3, 6)9, (4, 6)4, (4, 7)20, (5, 6)18, (6, 7)25\};$$

按照普里姆算法从顶点 1 出发得到最小生成树，试写出在最小生成树中依次得到的各条边。

四、阅读算法（每题 7 分，共 14 分）

```

1. void AE(Stack& S){
    InitStack(S);
    Push(S, 3);
    Push(S, 4);
    int x=Pop(S)+2*Pop(S);
    Push(S, x);
    int i, a[5]={1, 5, 8, 12, 15};
    for(i=0; i<5; i++) Push(S, 2*a[i]);
    while(!StackEmpty(S)) cout<<Pop(S)<<' ' ;
}
  
```

该算法被调用后得到的输出结果为：

```

2. void ABC (BTNode *BT,int &c1,int &c2) {
    if (BT != NULL) {
        ABC(BT->left,c1,c2);
        c1++;
        if (BT->left==NULL&&BT->right==NULL) c2++;
        ABC(BT->right,c1,c2);
    }/if
}
  
```

该函数执行的功能是什么？

五、 算法填空（共 8 分）

向单链表的末尾添加一个元素的算法。

```
Void InsertRear(LNode*& HL, const ElemType& item)
{
    LNode* newptr;
    newptr=new LNode;
    If (_____)
    {
        cerr<<"Memory allocation failare!"<<endl;
        exit(1);
    }
    _____=item;
    newptr->next=NULL;
    if (HL==NULL)
        HL=_____ ;
    else{
        LNode* P=HL;
        While (P->next!=NULL)
            _____;
        p->next=newptr;
    }
}
```

六、 编写算法（共 8 分）

编写从类型为 List 的线性表 L 中将第 i 个元素删除的算法，（假定不需要对 i 的值进行有效性检查，也不用判别 L 是否为空表。）

```
void Delete(List& L, int i)
```

模拟试卷一参考答案

一、 单选题（每题 2 分，共 20 分）

1.B 2.D 3.A 4.B 5.B 6.C 7.A 8.C 9.B 10.B

二、 填空题（每空 1 分，共 26 分）

- 顺序 链表 索引 散列
- $O(n)$ $O(1)$
- $p \rightarrow next = HS; HS = p$ $HS = HS \rightarrow next$
- $2i$ $2i+1$ $\lfloor i/2 \rfloor$ (或 $i/2$)

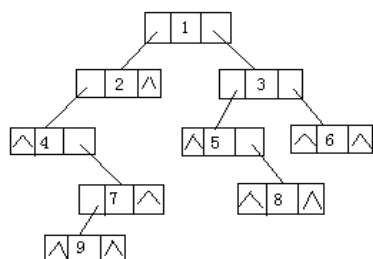


图 3

- 向上 根
- 2.9
- 邻接矩阵 邻接表 边集数组
- 1 4
- $O(n)$ $O(n \log_2 n)$ $O(n)$
- $\lceil m/2 \rceil - 1$ $m-1$ $\lceil m/2 \rceil$ m

三、 运算题（每题 6 分，共 24 分）

- (1) 3 X * Y 2 - / 1 +
 - (2) 2 X Y 3 + * +
- (1)

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|-----|----|-----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | ... | 18 | ... | 31 |
| | 1 | 2 | 3 | 4 | | 5 | 6 | | 7 | | | | 8 | | ... | 9 | | |

(2) 见图 3 所示：

- (1)不是小根堆。调整为：{12,24,33,65,34,56,48,92,86,70}
 - (2)是小根堆。

4. 普里姆算法从顶点 1 出发得到最小生成树为：
(1, 2) 3, (1, 3) 5, (1, 4) 8, (4, 6) 4, (2, 5) 10, (4, 7) 20

一、 阅读算法（每题 7 分，共 14 分）

1. 30 24 16 10 2 10
2. 该函数的功能是：统计出 BT 所指向的二叉树的结点总数和叶子总数

二、 算法填空（共 8 分，每一空 2 分）

newptr==NULL newptr->=data newptr p=p->next

三、 编写算法（8 分）

```
void Delete(List& L, int i)
{
    for(int j=i-1; j<L.size-1; j++)
        L.list[j]=L.list[j+1];           //第 i 个元素的下标为 i-1
    L.size--;
}
```

模拟试卷二

一、 单选题（每题 2 分，共 20 分）

- 在一个带有附加表头结点的单链表 HL 中，若要向表头插入一个由指针 p 指向的结点，则执行()。
A. HL=p; p->next=HL; B. p->next=HL->next; HL->next=p;
C. p->next=HL; p=HL; D. p->next=HL; HL=p;
- 若顺序存储的循环队列的 QueueMaxSize=n, 则该队列最多可存储() 个元素。
A. n B. n-1
C. n+1 D. 不确定
- 下述哪一条是顺序存储方式的优点？()
A. 存储密度大 B. 插入和删除运算方便
C. 获取符合某种条件的元素方便 D. 查找运算速度快
- 设有一个二维数组 A[m][n], 假设 A[0][0] 存放在位置 600₍₁₀₎, A[3][3] 存放在位置 678₍₁₀₎, 每个元素占一个空间, 问 A[2][3]₍₁₀₎ 存放在什么位置？(脚注₍₁₀₎表示用 10 进制表示, m>3)
A. 658 B. 648 C. 633 D. 653
- 下列关于二叉树遍历的叙述中，正确的是()。
A. 若一个树叶是某二叉树的中序遍历的最后一个结点，则它必是该二叉树的前序遍历最后一个结点
B. 若一个点是某二叉树的前序遍历最后一个结点，则它必是该二叉树的中序遍历的最后一个结点
C. 若一个结点是某二叉树的中序遍历的最后一个结点，则它必是该二叉树的前序最后一个结点
D. 若一个树叶是某二叉树的前序最后一个结点，则它必是该二叉树的中序遍历最后一个结点
- k 层二叉树的结点总数最多为()。
A. 2^k-1 B. $2K+1$ C. $2K-1$ D. 2^{k-1}
- 对线性表进行二分法查找，其前提条件是()。
A. 线性表以链接方式存储，并且按关键码值排好序
B. 线性表以顺序方式存储，并且按关键码值的检索频率排好序
C. 线性表以顺序方式存储，并且按关键码值排好序
D. 线性表以链接方式存储，并且按关键码值的检索频率排好序
- 对 n 个记录进行堆排序，所需要的辅助存储空间为
A. $O(\log_2 n)$ B. $O(n)$ C. $O(1)$ D. $O(n^2)$
- 对于线性表 (7, 34, 77, 25, 64, 49, 20, 14) 进行散列存储时，若选用 $H(K) = K \% 7$ 作为散列函数，则散列地址为 0 的元素有() 个，

- A. 1 B. 2 C. 3 D. 4

10. 下列关于数据结构的叙述中，正确的是()。

- A. 数组是不同类型值的集合
B. 递归算法的程序结构比迭代算法的程序结构更为精炼
C. 树是一种线性结构
D. 用一维数组存储一棵完全二叉树是有效的存储方法

二、填空题（每空 1 分，共 26 分）

- 数据的逻辑结构被分为_____、_____、_____和_____四种。
- 一个算法的时间复杂度为 $(3n^3+2000n\log_2n+90)/n^2$ ，其数量级表示为_____。
- 对于一个长度为 n 的单链存储的队列，在表头插入元素的时间复杂度为_____，在表尾插入元素的时间复杂度为_____。
- 假定一棵树的广义表表示为 $A(D(E, G), H(I, J))$ ，则树中所含的结点数为_____个，树的深度为_____，树的度为_____。
- 后缀算式 $79 \ 2 \ 30 \ + \ - \ 4 \ 2 \ /\ *$ 的值为_____。中缀算式 $(3+X*Y) - 2Y/3$ 对应的后缀算式为_____。
- 在一棵高度为 5 的理想平衡树中，最少含有_____个结点，最多含有_____个结点。
- 在树中，一个结点的直接后继结点称为该结点的_____。一个结点的直接前趋结点称为该结点的_____。
- 在一个具有 10 个顶点的无向完全图中，包含有_____条边，在一个具有 n 个顶点的有向完全图中，包含有_____条边。
- 假定一个线性表为(12,17,74,5,63,49,82,36)，若按 $\text{Key} \% 4$ 条件进行划分，使得同一余数的元素成为一个子表，则得到的四个子表分别为_____、_____、_____和_____。
- 对一棵 B_树进行删除元素的过程中，若最终引起树根结点的合并时，会使新树的高度比原树的高度_____。
- 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_____。
- 在线性表的散列存储中，装填因子 α 又称为装填系数，若用 m 表示散列表的长度， n 表示待散列存储的元素的个数，则 α 等于_____。

三、运算题（每题 6 分，共 24 分）

- 在如下数组 A 中链接存储了一个线性表，表头指针存放在 $A[0].\text{next}$ ，试写出该线性表。

| | | | | | | | | |
|------|---|----|----|----|----|----|---|----|
| A | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| data | | 60 | 50 | 78 | 90 | 34 | | 40 |
| next | 4 | 0 | 5 | 2 | 7 | 1 | | 3 |

- 已知一棵二叉树的前序遍历的结果是 ABKCDFGHIJ，中序遍历的结果是 KBCDAFHIGJ，试画出这棵二叉树。

- 已知一个图的顶点集 V 为： $V=\{1,2,3,4,5,6,7\}$;

其共有 10 条边。该图用如下边集数组存储：

| | | | | | | | | | | |
|----|---|---|---|---|---|---|---|---|---|---|
| 起点 | 1 | 2 | 2 | 5 | 5 | 2 | 2 | 6 | 1 | 3 |
| 终点 | 6 | 4 | 5 | 4 | 7 | 6 | 7 | 7 | 7 | 5 |
| 权 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 4 | 5 | 7 |

试用克鲁斯卡尔算法依次求出该图的最小生成树中所得到的各条边及权值。

- 画出向小根堆中加入数据 4, 2, 5, 8, 3, 6, 10, 1 时，每加入一个数据后堆的变化。

四、阅读算法（每题 7 分，共 14 分）

- 在下面的每个程序段中，假定线性表 La 的类型为 List，元素类型 ElemType 为 int，并假定每个程序段是连续执行的。试写出每个程序段执行后所得到的线性表 La。

```
(1) InitList(La);
    Int a[]={100, 26, 57, 34, 79};
    For (i=0; i<5; i++)
        Insert(La, a[i]);
```

```

    TraverseList(La);
(2) DeleteFront(La);
    InsertRear(La, DeleteFront(La));
    TraverseList(La);
(3) ClearList(La);
    For (i=0;i<5;i++)
        InsertFront(La, a[i]);
    TraverseList(La);
2. 现面算法的功能是什么?
void ABC(BTNode * BT)
{
    if BT {
        cout<<BT->data<<' ';
        ABC(BT->left);
        ABC(BT->right);
    }
}

```

五、 算法填空 (共 8 分)

二分查找的递归算法。

```

Int Binsch(ElemType A[], int low, int high, KeyType K)
{
    if _____{
        int mid=(low+high)/2;
        if (_____) return mid;    //查找成功, 返回元素的下标
        else if (K<A[mid].key)
            return Binsch(A, low, mid-1, K);    //在左子表上继续查找
        else return _____;    //在右子表上继续查找
    }
    else _____;    //查找失败, 返回-1
}

```

六、 编写算法 (共 8 分)

HL 为单链表的表头指针, 试写出在该单链表中查找具有给定的元素 item 的算法。

```
bool Find(LNode* HL, ElemType &item)
```

模拟试卷二参考答案

一、 单选题 (每题 2 分, 共 20 分)

1.B 2.B 3.A 4.D 5.A 6.A 7.C 8.C 9.D 10.D

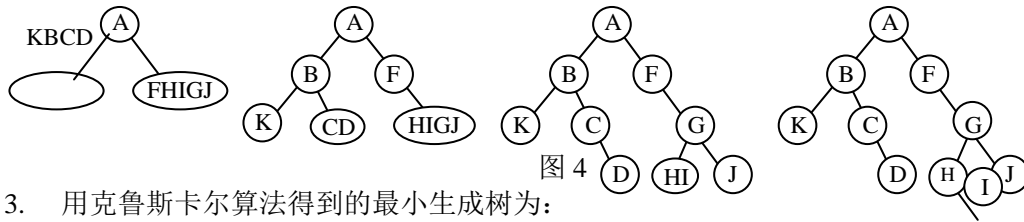
二、 填空题 (每空 1 分, 共 26 分)

- 集合结构 线性结构 树结构 图结构
- $O(n)$
- $O(1)$ $O(1)$
- 7 3 2
- 94 3 $X Y * + 2 Y * 3 / -$
- 16 31
- 孩子 (或子) 结点 双亲 (或父) 结点
- 45 $n(n-1)$
- (12, 36) (17, 5, 49) (74, 82) (63)
- 减少 1 (或减少)
- $O(\log_2 n)$ $O(n \log_2 n)$
- n/m

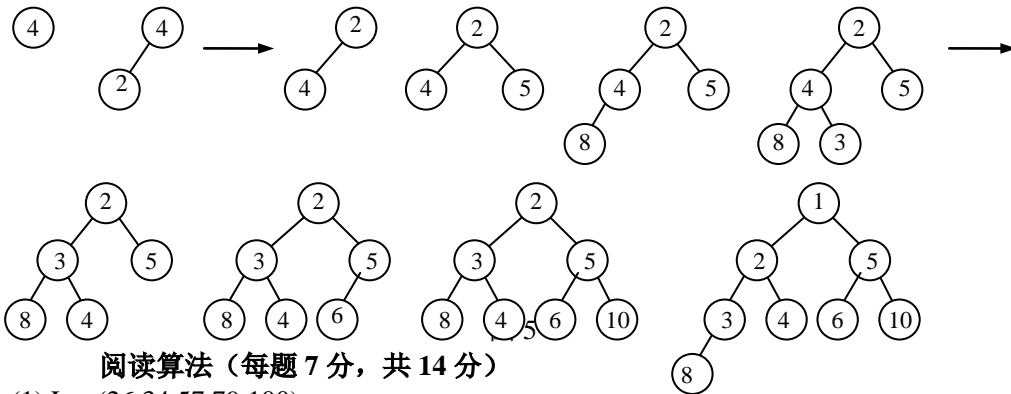
三、 运算题 (每题 6 分, 共 24 分)

- 线性表为: (90, 40, 78, 50, 34, 60)

2. 当前序序列为 ABKCDFGHIJ，中序序列为 KBCDAFHIGJ 时，逐步形成二叉树的过程如下图 4 所示：



3. 用克鲁斯卡尔算法得到的最小生成树为：
 (1,6)1, (2,4)1, (2,5)2, (5,7)2, (2,6)3, (3,5)7
4. 见图 5。



四、阅读算法（每题 7 分，共 14 分）

1. (1) La=(26,34,57,79,100)
 (2) La=(57,79,100,34)
 (3) La=(79,34,57,26,100)

2. 前序遍历链式存储的二叉树。

五、算法填空（每空 2 分，共 8 分）

(low<=high) K==A[mid].key Binsch(A, mid+1, high, K) return -1

六、编写算法（8 分）

```
bool Find(LNode* HL, ElemType &item)
{
    LNode* p=HL;
    while p
        if (p->data==item){
            return true;
        }
        else p=p->next;
    return false;
}
```

数据结构模拟试题 3

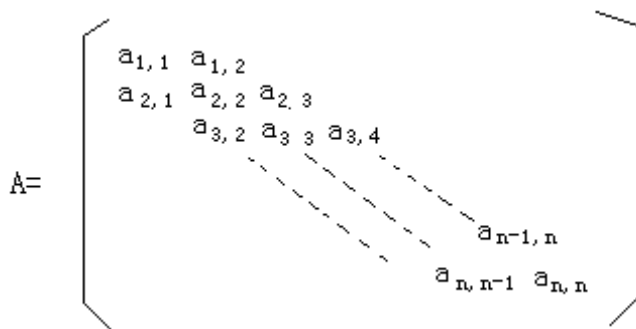
一、选择题（20 分）

1. 数据序列 (2, 1, 4, 9, 8, 10, 6, 20) 只能是下列排序算法中的 () 的两趟排序后的结果
 A) 快速排序 B) 冒泡排序
 C) 选择排序 D) 插入排序
2. 分别以下列序列构造二叉排序树，与用其它三个序列所构造的结果不同的是 ()
 A) (100, 80, 90, 60, 120, 110, 130)
 B) (100, 120, 110, 130, 80, 60, 90,)

- C) (100, 60, 80, 90, 120, 110, 130)
 D) (100, 80, 60, 90, 120, 130, 110)
3. 一棵左右子树均不空的二叉树先序线索化后, 其中空链域的个数是 ()
 A) 0 B) 1 C) 2 D) 不确定
4. 设一个栈的输入序列是 1, 2, 3, 4, 5, 则下列序列中, 是合法输出序列的是 ()
 A) 5 1 2 3 4 B) 4 5 1 3 2
 C) 4 3 1 2 5 D) 3 2 1 5 4
5. 在图采用邻接表存储时, 求最小生成树的 prim 算法的时间复杂度为 ()
 A) $O(N)$ B) $O(N+E)$ C) $O(N^2)$ D) $O(N^3)$
6. 下列排序中, () 是堆。
 A) (100, 80, 55, 60, 50, 40, 58, 35, 20)
 B) (100, 80, 55, 60, 50, 40, 35, 58, 20)
 C) (100, 80, 55, 58, 50, 40, 60, 35, 20)
 D) (100, 70, 55, 60, 50, 40, 58, 35, 20)
7. 设森林 F 对应的二叉树为 B, 它有 m 个结点, B 的根为 p, p 的右子树结点个数为 n, 森林 F 中第一棵树的结点个数是 ()
 A) m-n B) m-n-1 C) n+1 D) 条件不足, 无法确定
8. 采用简单选择排序, 比较次数与移动次数分别为 ()
 A) $O(n)$, $O(\log n)$ B) $O(\log n)$, $O(n*n)$
 C) $O(n*n)$, $O(n)$ D) $O(n\log n)$, $O(n)$
9. 已知广义表 $L = (((a)), ((b)), (c), d)$, 利用 head 和 tail 运算把原子项 c 从 L 中分离出来的表达式为 ()
 A) head(tail(head(tail(L))))
 B) head(head(tail(tail(L))))
 C) tail(head(head(tail(L))))
 D) tail(tail(head(head(L))))
10. 设有一个按元素值排好序的线性表且长度大于 2, 对给定的值 k, 分别用顺序查找法和二分查找法查找一个与 k 相等的元素, 比较次数分别是 s 和 b, 在查找不成功的情况下, 正确的 s 和 b 的数量关系是 ()
 A) 总有 $s=b$ B) 总有 $s>b$
 C) 总有 $s<b$ D) 与 k 值大小有关

二、填空题 (20 分)

- 已知完全二叉树有 30 个结点, 则整个二叉树有_____个度为 0 的结点。
- 有数据 $WG = \{7, 19, 2, 6, 32, 3, 21, 10\}$, 则所建 Huffman 树的树高是_____, 带权路径长度 WPL 为_____。
- 设有三对角矩阵如下图



将带状区中的元素 a_{ij} ($|i-j| \leq 1$) 放在一维数组 B 中, 则 B 的大小为_____, 元素 a_{ij} 在 B 中的位置是_____ (下标从 0 开始计)。

4. 按 13, 24, 37, 90, 53 的次序形成二叉平衡树, 则该二叉平衡树的根为_____, 左子树中的数据是_____, 右子树中的数据是_____。

5. 下面程序段中带下划线的语句的执行次数的数量级是_____。

```
i=1;
while (i<n) i=i*2;
```

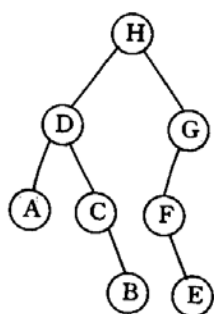
6. 无向图 $G(V, E)$, 其中 $V(G) = \{1, 2, 3, 4, 5, 6, 7\}$, $E(G) = \{(1, 2), (1, 3), (2, 4), (2, 5), (3, 6), (3, 7), (6, 7), (5, 1)\}$, 对该图从顶点 3 开始进行遍历, 去掉遍历中未走过的边, 得一生成树 $G'(V, E')$, $V(G') = V(G)$, $E(G') = \{(1, 3), (3, 6), (7, 3), (1, 2), (1, 5), (2, 4)\}$, 则采用的遍历方法是_____。

三、判断题 (10 分)

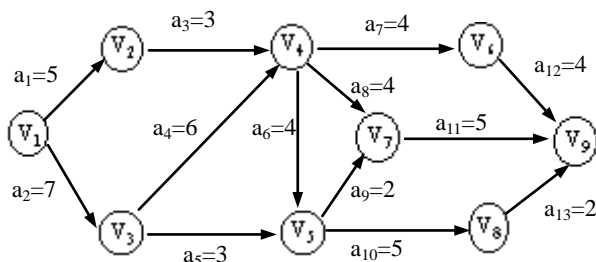
1. 存在这样的二叉树, 对它采用任何次序的遍历, 结果相同。()
2. 队列和栈都是运算受限的线性表, 只允许在表的两端进行运算。()
3. 非空的广义表的表尾只能是一个子表而不可能是一个原子。()
4. 有向图的邻接表和逆邻接表中的结点的个数可能不等。()
5. 堆是一个完全二叉树, 反之亦然。()
6. 链表是采用链式存储结构的线性表, 进行插入、删除操作时, 在链表中比在顺序存储结构中效率高。()
7. 在图 G 的最小生成树中, 可能会有某条边的权值超过未选边的权值。()
8. 二叉树中除叶结点外, 任一结点 x , 其左子树根结点的值小于该结点 (x) 的值, 其右子树根结点的值大于该结点 (x) 的值, 则此二叉树一定是二叉排序树。()
9. 用二叉链表存储 n 个结点的二叉树, 结点的 $2n$ 个指针域中有 $n-1$ 个空指针。()
10. 给定一棵树, 可以找到唯一的一棵二叉树与之对应。()

四、应用题 (20 分)

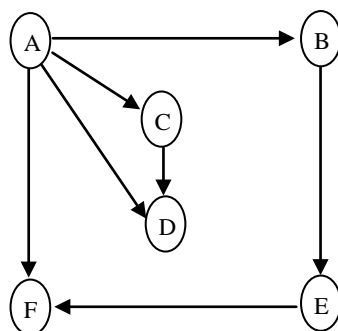
1. 对下图所示二叉树分别按先序、中序、后序遍历, 给出相应的结点序列, 同时给二叉树加上后序后继线索。



2. 给出下图所示 AOE-网中的关键路径。

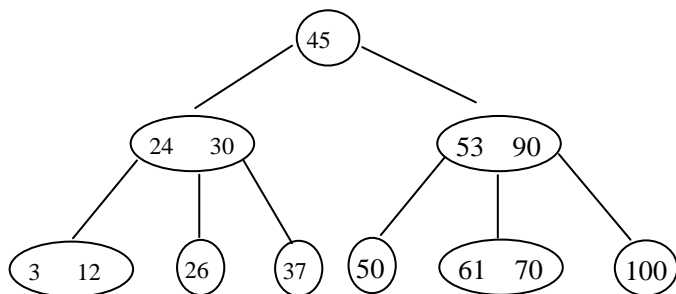


3. 对下图所示有向图，列出 4 种可能的拓扑有序序列。



4. 设一组数据为{1, 14, 27, 29, 55, 68, 10, 11, 23}，现采用的哈希函数是 $H(\text{key}) = \text{key} \% 13$ ，冲突用链地址法解决，设哈希表的大小为 13 (0..12)，试画出插入上述数据后的哈希表。

5. 已知 2-3 树如图所示，当插入一个数 85 后，画出调整后的 2-3 树。



6. 对给定文件 (28, 07, 39, 10, 65, 14, 61, 17, 50, 21)，选择第一个元素 28 进行划分，写出其快速排序第一趟的排序过程。

五、算法设计题 (30 分)

1. 请设计一个算法，要求该算法把二叉树的叶子结点按从左到右的顺序连成一个单链表，表头指针为 head。二叉树按二叉链表方式存储，链接时用叶子结点的右指针域来存放单链表指针 (10 分)。
2. 一个无向连通图的存储结构以邻接表的形式给定，删除该图中的一条边 (i, j)。叙述思

路并编写算法（10 分）。

3. 线性表 $(a_1, a_2, a_3, \dots, a_n)$ 中元素递增有序且按顺序存于计算机内（10 分）。要求设计一算法完成：

- 1) 用最少的时间在表中查找数值为 x 的元素。
- 2) 若找不到将其插入表中并使表中元素仍递增有序。

数据结构模拟试题 3 参考答案

一、选择题（20 分）

1-5 A C B D C

6-10 B A C B D

二、填空题（20 分）

1. 15
2. 6 261
3. $3n-2$ $2i+j-3$
4. 24 13 53
5. 对数阶 ($O(\log_2^N)$)
6. 广度优先遍历

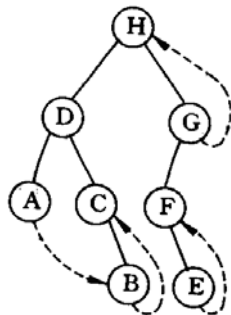
三、判断题（10 分）

- | | | | | |
|-----------------|-----------------|-----------------|-------------|------------------|
| 1. \checkmark | 2. \times | 3. \checkmark | 4. \times | 5. \times |
| 6. \checkmark | 7. \checkmark | 8. \times | 9. \times | 10. \checkmark |

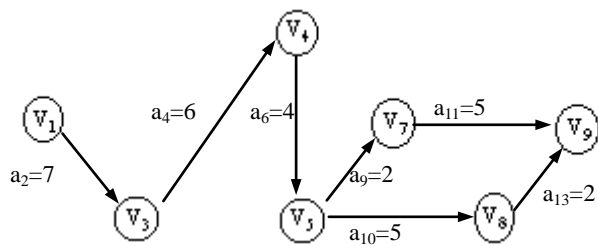
四、应用题（20 分）

1. 参考答案如下：

| | |
|--------|----------|
| 先序遍历序列 | HDACBGFE |
| 中序遍历序列 | ADCBFEGH |
| 后序遍历序列 | ABCDEFHG |



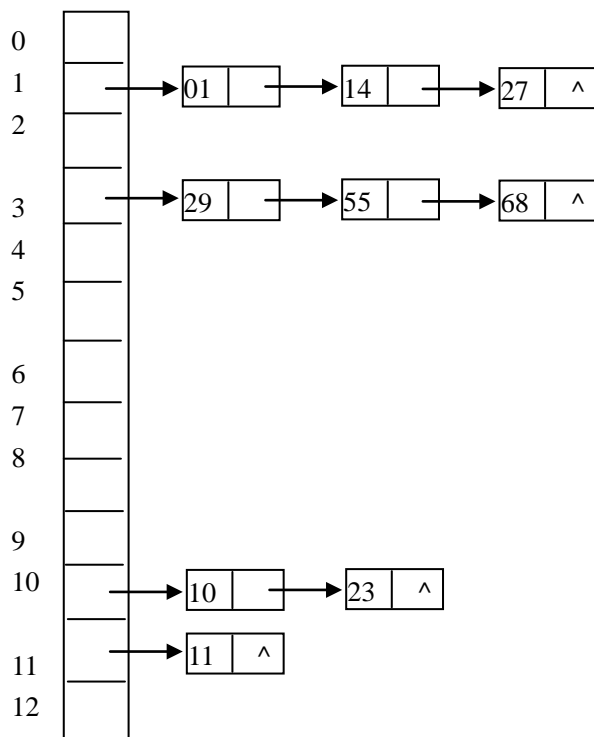
2. 参考答案如下：



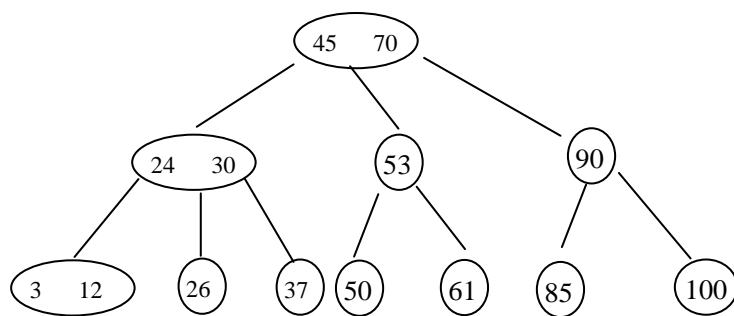
3. 参考答案如下:

- 1) A B C D E F
- 2) A B C E D F
- 3) A B E F C D
- 4) A C D B E F

4. 参考答案如下:



5. 参考答案如下:



6. 参考答案如下:

初始序列: 28, 07, 39, 10, 65, 14, 61, 17, 50, 21

21 前移 : 21, 07, 39, 10, 65, 14, 61, 17, 50, []

39 后移 : 21, 07, [], 10, 65, 14, 61, 17, 50, 39

17 前移 : 21, 07, 17, 10, 65, 14, 61, [], 50, 39

65 后移 : 21, 07, 17, 10, [], 14, 61, 65, 50, 39

14 前移 : 21, 07, 17, 10, 14, [], 61, 65, 50, 39

排序结果: {21, 07, 17, 10, 14} 28 {61, 65, 50, 39}

五. 算法设计题 (30 分)

1. 算法源代码如下:

```
bitree head,tail;
void leaf_link(bitree T)
/*编写函数将二叉树的所有叶子结点从左到右链成一个单链表 head*/
if(T)
{ leaf_link(T->lchild);
  if(!T->lchild && !T->rchild)
  { if(head==NULL){head=T;tail=T;}
    else {tail->rchild=T;tail=T;}
  }
  leaf_link(T->rchild);
}
```

2. 算法源代码如下:

```
int delete(algraph graph,int i,int j)
{ arcptr q, p;
/*删除顶点  $V_i$  的邻接点链表中的表结点  $j$ */
p=graph.vertices[i].firstarc;
while(p!=NULL&& p->adjvex!=j)
{ q=p; p=p->nextarc;}
if(p->adjvex==j)
q->netarc=p->nextarc;
/*删除顶点  $V_j$  的邻接点链表中的表结点  $i$ */
p=graph.vertices[j].firstarc;
while(p!=NULL&& p->adjvex!=i)
{ q=p; p=p->nextarc;}
if(p->adjvex==i)
q->netarc=p->nextarc;
}
```

3. 算法源代码如下:

```
void binsert_sort(stable st, int x)
/*对有序表 st 进行动态折半查找*/
int i,j,low,high,mid;
if(x== st.elem[0]||x== st.elem[st.length]) return;
```

```

else
{
    st.elem[0]=x;
    low=1;high= st.length;
    while(low<=high) /*折半查找插入位置 low*/
    {
        mid=(low+high)/2;
        if(st.elem[0]==st.elem[mid]) return;
        else if(st.elem[0]<st.elem[mid]) high=mid-1;
        else low=mid+1; }
    for(j= st.length;j>=low;j--) st.elem[j+1]=st.elem[j];/*后移元素*/
    st.elem[low]=st.elem[0];
    st.length++;
}
}

```

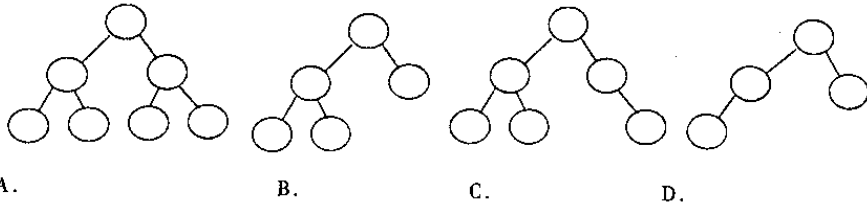
模拟试卷四

一、 单选题（每题 2 分，共 20 分）

- 以下数据结构中哪一个是线性结构？（ ）
A. 有向图 B. 栈 C. 二叉树 D. B 树
- 若某链表最常用的操作是在最后一个结点之后插入一个结点和删除最后一个结点，则采用（ ）存储方式最节省时间。
A. 单链表 B. 双链表
C. 带头结点的双循环链表 D. 单循环链表
- 以下哪一个不是队列的基本运算？（ ）
A. 在队列第 i 个元素之后插入一个元素 B. 从队头删除一个元素
C. 判断一个队列是否为空 D. 读取队头元素的值
- 字符 A、B、C、D 依次进入一个栈，按出栈的先后顺序组成不同的字符串，至多可以组成（ ）个不同的字符串？
A. 15 B. 14 C. 16 D. 21
- 由权值分别为 4,7,6,2 的叶子生成一棵哈夫曼树，它的带权路径长度为（ ）。
A. 11 B. 37 C. 19 D. 53

以下 6-8 题基于下面的叙述：若某二叉树结点的中序遍历的序列为 A、B、C、D、E、F、G，后序遍历的序列为 B、D、C、A、F、G、E。

- 则该二叉树结点的前序遍历的序列为（ ）。
A. E、G、F、A、C、D、B B. E、A、G、C、F、B、D
C. E、A、C、B、D、G、F D. E、G、A、C、D、F、B
- 该二叉树有（ ）个叶子。
A. 3 B. 2 C. 5 D. 4
- 该二叉树的按层遍历的序列为（ ）
A. E、G、F、A、C、D、B B. E、A、C、B、D、G、F
C. E、A、G、C、F、B、D D. E、G、A、C、D、F、B
- 下面的二叉树中，（ ）不是完全二叉树。



10. 设有关键码序列(q, g, m, z, a), 下面哪一个序列是从上述序列出发建的小根堆的结果?()
- A. a, g, m, q, z B. a, g, m, z, q
C. g, m, q, a, z D. g, m, a, q, z

二、填空题（每空 1 分，共 26 分）

1. 数据结构是指数据及其相互之间的_____。当结点之间存在 1 对 N (1: N) 的联系时, 称这种结构为_____。
2. 一个广义表中的元素分为_____元素和_____元素两类。
3. 对于一个长度为 n 的顺序存储的线性表, 在表头插入元素的时间复杂度为_____, 在表尾插入元素的时间复杂度为_____。
4. 向一个由 HS 指向的链栈中插入一个结点时 p 时, 需要执行的操作是_____；删除一个结点时, 需要执行的操作是_____（假设栈不空而且无需回收被删除结点）。
5. 栈又称为_____表, 队列又称为_____表。
6. 在稀疏矩阵所对应的三元组线性表中, 每个三元组元素按_____为主序、_____为辅序的次序排列。
7. 若一棵二叉树中只有叶子结点和左、右子树皆非空的结点, 设叶结点的个数为 K, 则左、右子树皆非空的结点个数是_____。
8. 以折半（或二分）查找方法从长度为 8 的有序表中查找一个元素时, 平均查找长度为_____。
9. 表示图的三种常用的存储结构为_____、_____和_____。
10. 对于线性表 (78, 4, 56, 30, 65) 进行散列存储时, 若选用 $H(K) = K \% 5$ 作为散列函数, 则散列地址为 0 的元素有_____个, 散列地址为 4 的有_____个。
11. 在归并排序中, 进行每趟归并的时间复杂度为_____, 整个排序过程的时间复杂度为_____, 空间复杂度为_____。
12. 在 n 个带权叶子结点构造出的所有二叉树中, 带权路径长度最小的二叉树称为_____。WPL 称为_____。
13. 在索引表中, 若一个索引项对应主表的一个记录, 则此索引为_____索引, 若对应主表的若干条记录, 则称此索引为_____索引。

三、运算题（每题 6 分，共 24 分）

1. 写出下列中缀表达式的后缀形式：
 - (1) $3X/(Y-2H)+1$
 - (2) $2+X*(Y+3)$
2. 假定一棵二叉树广义表表示为 a(b(c),d(e,f)), 分别写出对它进行先序、中序、后序、按层遍历的结果。

先序:

中序:

后序:

按层:

3. 已知一个无向图的顶点集为{a, b, c, d, e},其邻接矩阵如下所示

| | | | | | |
|---|---|---|---|---|---|
| a | 0 | 1 | 0 | 0 | 1 |
| b | 1 | 0 | 0 | 1 | 0 |
| c | 0 | 0 | 0 | 1 | 1 |
| d | 0 | 1 | 1 | 0 | 1 |
| e | 1 | 0 | 1 | 1 | 0 |

- (1) 画出该图的图形;
 (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. 已知一个图的顶点集 V 和边集 E 分别为:
 $V = \{0, 1, 2, 3, 4, 5, 6, 7\}$;
 $E = \{(0, 1)8, (0, 2)5, (0, 3)2, (1, 5)6, (2, 3)25, (2, 4)13, (3, 5)9, (3, 6)10, (4, 6)4, (5, 7)20\}$;

按照普里姆算法从顶点 0 出发得到最小生成树, 试写出在最小生成树中依次得到的各条边。

四、 阅读算法 (每题 7 分, 共 14 分)

```
1. void AE(Stack& S){
    InitStack(S);
    Push(S, 3);
    Push(S, 4);
    int x=Pop(S)+2*Pop(S);
    Push(S, x);
    int i, a[5]={2, 5, 8, 22, 15};
    for(i=0; i<5; i++) Push(S, a[i]);
    while(!StackEmpty(S)) cout<<Pop(S)<<' ' ;
}
```

该算法被调用后得到的输出结果为:

```
2. int akm ( unsigned m, unsigned n ) {
    if ( m == 0 ) return n+1;
    else if ( n == 0 ) return akm ( m-1, 1 );
    else return akm ( m-1, akm ( m, n-1 ) );
}
```

该函数执行的功能是什么?

五、 算法填空 (共 8 分)

二叉搜索树的查找—非递归算法

```
bool Find(BTreeNode* BST, ElemType& item)
{while(BST!=NULL) {
    if (item==_____) {
        item=BST->data; //查找成功
        return true;}
    else if(item<BST->data)
        BST=BST->_____;
    else BST=BST->_____;
} //while
return _____; //查找失败
}
```

六、 编写算法 (共 8 分)

用递归的算法编写出对存入在 a[n+1] 数组中的 n 个有序元素进行二分 (又称折半) 查找

(假定 a[0] 单元不用) 的程序。

```
int halfsearch(SSTable *a, KeyType k, int low, int high)
```

模拟试卷四参考答案

一、单选题 (每题 2 分, 共 20 分)

1.B 2.C 3.A 4.B 5.B 6.C 7.A 8.C 9.C 10.B

二、填空题 (每空 1 分, 共 26 分)

1. 联系 树 (或树结构)
2. 单 (子) 表
3. $O(n)$ $O(1)$
4. $p \rightarrow next = HS; HS = p$ $HS = HS \rightarrow next$
5. 先进后出 先进先出
6. 行 列
7. $k-1$
8. 2.625
9. 邻接矩阵 邻接表 边集数组
10. 2 1
11. $O(n)$ $O(n \log_2 n)$ $O(n)$
12. 哈夫曼树 带权路径长度
13. 稠密 稀疏

三、运算题 (每题 6 分, 共 24 分)

1. (1) $3 \ X \ * \ Y \ 2 \ H \ * \ / \ 1 \ +$
(2) $2 \ X \ Y \ 3 \ + \ * \ +$
2. 先序: a,b,c,d,e,f
中序: c, b, a, e, d, f
后序: c, b, e, f, d, a
按层: a,b,d,c,e,f
3. (1) 该图的图形如图 9 示:
(2) 深度优先遍历序列为: abdcce
广度优先遍历序列为: abedc
4. 普里姆: (0,3)2, (0,2)5, (0,1)8, (1,5)6, (3,6)10, (6,4)4, (5,7)20

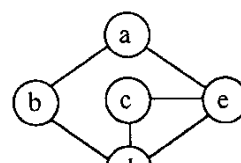


图 9

四、阅读算法 (每题 7 分, 共 14 分)

3. 15 22 8 5 2 10
4. 该函数的功能是:

$$akm(m, n) = \begin{cases} n+1 & \text{当 } m=0 \text{ 时} \\ akm(m-1, 1) & \text{当 } m \neq 0, n=0 \text{ 时} \\ akm(m-1, akm(m, n-1)) & \text{当 } m \neq 0, n \neq 0 \text{ 时} \end{cases}$$

求:

五、算法填空 (共 8 分, 每一空 2 分)

BST \rightarrow data left right false

六、编写算法 (8 分)

递归算法:

```
int halfsearch(SSTable *a, KeyType k, int low, int high)
{ if (low > high)
    return 0;
  else {
    int mid = (low + high) / 2;
    if EQ(k, a[mid].key) return mid;
    else if LT(k, a[mid].key) return halfsearch(a, k, low, mid - 1);
    else return halfsearch(a, k, mid + 1, high);
  }
```

}

模拟试卷五

一、单选题（每题 2 分，共 20 分）

1. 栈和队列的共同特点是()。
A. 只允许在端点处插入和删除元素
B. 都是先进后出
C. 都是先进先出
D. 没有共同点
2. 用链接方式存储的队列，在进行插入运算时()。
A. 仅修改头指针
B. 头、尾指针都要修改
C. 仅修改尾指针
D. 头、尾指针可能都要修改
3. 以下数据结构中哪一个是非线性结构？()
A. 队列
B. 栈
C. 线性表
D. 二叉树
4. 设有一个二维数组 $A[m][n]$ ，假设 $A[0][0]$ 存放位置在 $644_{(10)}$ ， $A[2][2]$ 存放位置在 $676_{(10)}$ ，每个元素占一个空间，问 $A[3][3]_{(10)}$ 存放在什么位置？脚注₍₁₀₎表示用 10 进制表示。
A. 688
B. 678
C. 692
D. 696
5. 树最适合用来表示()。
A. 有序数据元素
B. 无序数据元素
C. 元素之间具有分支层次关系的数据
D. 元素之间无联系的数据
6. 二叉树的第 k 层的结点数最多为()。
A. $2^k - 1$
B. $2K + 1$
C. $2K - 1$
D. 2^{k-1}
7. 若有 18 个元素的有序表存放在一维数组 $A[19]$ 中，第一个元素放 $A[1]$ 中，现进行二分查找，则查找 $A[3]$ 的比较序列的下标依次为()
A. 1, 2, 3
B. 9, 5, 2, 3
C. 9, 5, 3
D. 9, 4, 2, 3
8. 对 n 个记录的文件进行快速排序，所需要的辅助存储空间大致为
A. $O(1)$
B. $O(n)$
C. $O(\log_2 n)$
D. $O(n^2)$
9. 对于线性表 (7, 34, 55, 25, 64, 46, 20, 10) 进行散列存储时，若选用 $H(K) = K \% 9$ 作为散列函数，则散列地址为 1 的元素有()个，
A. 1
B. 2
C. 3
D. 4
10. 设有 6 个结点的无向图，该图至少应有()条边才能确保是一个连通图。
A. 5
B. 6
C. 7
D. 8

二、填空题（每空 1 分，共 26 分）

1. 通常从四个方面评价算法的质量：_____、_____、_____和_____。
2. 一个算法的时间复杂度为 $(n^3 + n^2 \log_2 n + 14n)/n^2$ ，其数量级表示为_____。
3. 假定一棵树的广义表表示为 $A(C, D(E, F, G), H(I, J))$ ，则树中所含的结点数为_____个，树的深度为_____，树的度为_____。
4. 后缀算式 $9\ 2\ 3\ +\ 10\ 2\ /\ -$ 的值为_____。中缀算式 $(3+4X) - 2Y/3$ 对应的后缀算式为_____。
5. 若用链表存储一棵二叉树时，每个结点除数据域外，还有指向左孩子和右孩子的两个指针。在这种存储结构中， n 个结点的二叉树共有_____个指针域，其中有_____个指针域是存放了地址，有_____个指针是空指针。
6. 对于一个具有 n 个顶点和 e 条边的有向图和无向图，在其对应的邻接表中，所含边结点分别有_____个和_____个。
7. AOV 网是一种_____的图。
8. 在一个具有 n 个顶点的无向完全图中，包含有_____条边，在一个具有 n 个顶点的有向完全图中，包含有_____条边。
9. 假定一个线性表为 (12, 23, 74, 55, 63, 40)，若按 $\text{Key} \% 4$ 条件进行划分，使得同一余数的元

素成为一个子表，则得到的四个子表分别为_____、
_____和_____。

10. 向一棵 B_树插入元素的过程中，若最终引起树根结点的分裂，则新树比原树的高度_____。

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. 在快速排序、堆排序、归并排序中，_____排序是稳定的。

三、运算题（每题 6 分，共 24 分）

1. 在如下数组 A 中链接存储了一个线性表，表头指针为 A[0].next，试写出该线性表。

| | | | | | | | | |
|------|---|----|----|----|----|----|---|----|
| A | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| data | | 60 | 50 | 78 | 90 | 34 | | 40 |
| next | 3 | 5 | 7 | 2 | 0 | 4 | | 1 |

2. 请画出图 10 的邻接矩阵和邻接表。

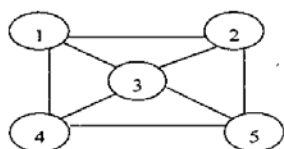


图 10

3. 已知一个图的顶点集 V 和边集 E 分别为：

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

$E = \{(1, 2) 3, (1, 3) 5, (1, 4) 8, (2, 5) 10, (2, 3) 6, (3, 4) 15, (3, 5) 12, (3, 6) 9, (4, 6) 4, (4, 7) 20, (5, 6) 18, (6, 7) 25\}$;

用克鲁斯卡尔算法得到最小生成树，试写出在最小生成树中依次得到的各条边。

4. 画出向小根堆中加入数据 4, 2, 5, 8, 3 时，每加入一个数

据后堆的变化。

四、阅读算法（每题 7 分，共 14 分）

1. LinkList mynote(LinkList L)

{//L 是不带头结点的单链表的头指针

if(L&&L->next){

q=L; L=L->next; p=L;

S1: while(p->next) p=p->next;

S2: p->next=q; q->next=NULL;

}

return L;

}

请回答下列问题：

(1) 说明语句 S1 的功能；

(2) 说明语句组 S2 的功能；

(3) 设链表表示的线性表为 (a_1, a_2, \dots, a_n) ，写出算法执行后的返回值所表示的线性表。

2. void ABC(BTNode * BT)

{

if BT {

ABC (BT->left);

ABC (BT->right);

cout<<BT->data<<' ';

}

}

该算法的功能是：

五、算法填空（共 8 分）

二叉搜索树的查找—递归算法：

bool Find(BTreeNode* BST, ElemType& item)

{

if (BST==NULL)

return false; //查找失败

```

else {
    if (item==BST->data){
        item=BST->data;//查找成功
        return _____;}
    else if(item<BST->data)
        return Find(_____, item);
    else return Find(_____, item);
} //if
}

```

六、 编写算法（共 8 分）

统计出单链表 HL 中结点的值等于给定值 X 的结点数。

```
int CountX(LNode* HL,ElemType x)
```

模拟试卷五参考答案

一、 单选题（每题 2 分，共 20 分）

1.A 2.D 3.D 4.C 5.C 6.D 7.D 8.C 9.D 10.A

二、 填空题（每空 1 分，共 26 分）

- 正确性 易读性 强壮性 高效率
- $O(n)$
- 9 3 3
- $-1 \quad 3 \quad 4 \quad X * + 2 \quad Y * 3 / -$
- $2n \quad n-1 \quad n+1$
- $e \quad 2e$
- 有向无回路
- $n(n-1)/2 \quad n(n-1)$
- (12, 40) () (74) (23, 55, 63)
- 增加 1
- $O(\log_2 n) \quad O(n \log_2 n)$
- 归并

三、 运算题（每题 6 分，共 24 分）

- 线性表为: (78, 50, 40, 60, 34, 90)

- 邻接矩阵:
$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

邻接表如图 11 所示:

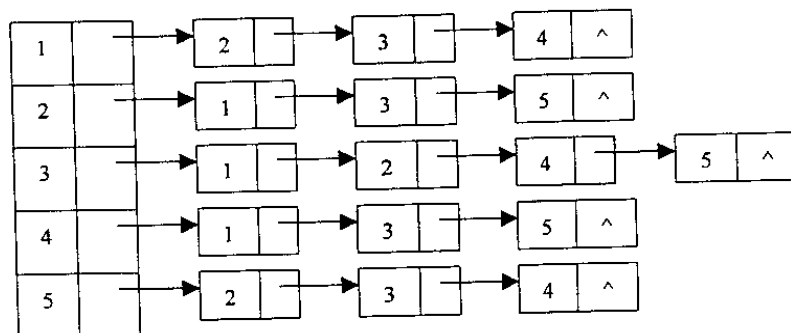
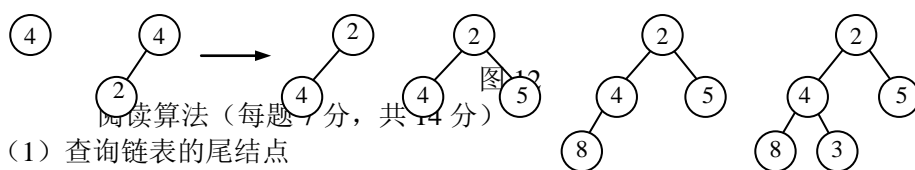


图 11

- 用克鲁斯卡尔算法得到的最小生成树为:
(1,2)3, (4,6)4, (1,3)5, (1,4)8, (2,5)10, (4,7)20

4. 见图 12



四、

1. (1) 查询链表的尾结点
(2) 将第一个结点链接到链表的尾部, 作为新的尾结点
(3) 返回的线性表为 $(a_2, a_3, \dots, a_n, a_1)$

2. 递归地后序遍历链式存储的二叉树。

五、 算法填空 (每空 2 分, 共 8 分)

true BST->left 8 4 BST->right

六、 编写算法 (8 分)

int CountX(LNode* HL, ElemType x)

```
{ int i=0; LNode* p=HL; //i 为计数器
  while(p!=NULL)
  { if (P->data==x) i++;
    p=p->next;
  } //while, 出循环时 i 中的值即为 x 结点个数
  return i;
} //CountX
```

数据结构模拟试题 (六)

本试卷分两部分, 第一部分为选择题, 第二部分为非选择题; 选择题 20 分, 非选择题 80 分, 满分 100 分。考试时间 150 分钟。

第一部分 选择题

一、单项选择题 (本大题共 20 小题, 每小题 1 分, 共 20 分)

在每小题列出的四个选项中只有一个选项是符合题目要求的, 请将正确选项前的字母填在题后的括号内。

1. 每一个存储节点不仅含有一个数据元素, 还包含一组指针, 该存储方式是【 】

- A. 顺序存储
- B. 链式存储
- C. 索引存储
- D. 散列存储

2. 下列算法的时间复杂度是【 】

```
for (i = 0; i < n; i++)
```

```
  c[i][j] = i + j;
```

- A. $O(1)$
- B. $O(n)$
- C. $O(\log_2 n)$
- D. $O(n^2)$

3. 线性表若采用链式存储结构时, 要求内存中可用存储单元的地址【 】

- A. 必须是连续的
- B. 部分地址必须是连续的
- C. 一定是不连续的
- D. 连续或不连续都可以

4. 在有 n 个结点的顺序表中做插入、删除运算, 需平均移动结点的数目为【 】

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

5. 在一个链队中, 假设 f 和 r 分别为队首和队尾指针, 则插入 s 所指节点的运算时【 】

- A. $f \rightarrow next = s; f = s;$
- B. $f \rightarrow next = s; r = s;$
- C. $s \rightarrow next = r; r = s;$
- D. $s \rightarrow next = f; f = s;$

6. 一个栈的入栈序列是 1, 2, 3, 4, 则栈的不可能的输出序列是:【 】

- A. 3, 4, 2, 1
- B. 3, 2, 4, 1
- C. 1, 2, 3, 4
- D. 4, 3, 1, 2

7. 设 $s3 = "IAM"$, $s4 = "A TEACHER"$, 则 $strcmp(s3, s4) =$ 【 】

- A. "IAM"
- B. "I AM A TEACHER"
- C. "I AMA TEACHER"
- D. "A TEACHER"

8. 设 $s3 = "IAM"$, $s4 = "A TEACHER"$, 则 $strcmp(s3, s4) =$ 【 】

1. 数据的_____结构是依赖于计算机的。
2. 稀疏索引中索引项的地址指示一组节点的_____存储位置。
3. 顺序存储是把线性表的节点按_____次序依次存放在一组地址连续的存储单元里。
4. 在一个单链表 p 所指节点之后插入一个 s 所指节点时, 应执行 $s \rightarrow next =$ _____和 $p \rightarrow next =$ _____。
5. 后进先出的线性表称为_____, 后进后出的线性表称为_____。
6. 顺序串上的串定位运算 NaiveStrMatch ("bbaabbcc", "aabb") 的值为_____。
7. n 阶三角矩阵的上三角元素值相等, 进行压缩存储时, 该值存储在下标为_____的数组元素中。
8. 广义表 $(a, (a, b)J, e, ((i, j), k))$ 的长度是_____, 深度是_____。

9. 深度为 k 的二叉树最多有_____个节点。
10. 树中节点的最大层次称为树的_____。
11. 用迪杰斯特拉 (Dijkstra) 算法求某一顶点到其余各顶点间的最短路径是按路径长度_____的次序来得到最短路径的。
12. 拓扑排序算法是通过重复选择具有_____个前趋顶点的过程来完成的。
13. 对于 n 个记录的集合进行归并排序, 所需要的附加空间是_____。
14. 在对一组记录 (54, 38, 96, 23, 15, 72, 60, 45, 83) 进行直接插入排序时, 当把第 7 个记录 60 插入到有序表时, 为寻找插入位置需比较_____次。
15. 对于长度为 n 的线性表, 若进行顺序查找, 则时间复杂度为_____; 若采用折半 (二分) 查找, 则时间复杂度为_____。
16. ISAM 的中文意思是_____, 采用_____索引结构。

三、名词解释 (本大题共 5 小题, 每题 3 分, 共 15 分)

1. 存储密度
2. 栈的“上溢”
3. 带状矩阵
4. 完全二叉树
5. 图

四、简答题 (本大题共 5 小题, 每题 5 分, 共 25 分)

1. 简述在单链表上尾指针取代头指针的作用。
2. 简述栈的基本运算。
3. 串的链式存储与串的顺序存储相比, 在哪些操作上效率更高?
4. 以数据集 {4, 5, 6, 7, 10, 12, 18} 为节点数值构造霍夫曼树。
5. 设数据序列为 $D = \{13, 28, 72, 5, 16, 8, 7, 9, 34\}$, 请为 D 组织散列表。散列函数为 $H(K) = K \% 7$, 散列表的长度为 10 个单元, 起始地址为 0, 要求用线性探测再散列来解决冲突。并计算查找成功的平均查找长度。

五、应用题 (本大题共 2 小题, 每题 10 分, 共 20 分)

1. 输入一字符串, 检查其中是否含有匹配的小括号。当括号匹配时, 按逆序输出括号中的字符, 否则输出错误信息。
2. 设有链式存储结构的二叉树, 计算其中有双后继节点的节点的个数。

数据结构模拟试题 (六) 参考答案

一、单选题答案

- | | | | | |
|-------|-------|-------|-------|-------|
| 1. B | 2. B | 3. D | 4. B | 5. B |
| 6. D | 7. D | 8. C | 9. D | 10. D |
| 11. B | 12. D | 13. B | 14. A | 15. B |
| 16. B | 17. D | 18. C | 19. B | 20. B |

二、填空题

- | | |
|--------------------------|---------------------------------|
| 1. 存储 | 2. 起始 |
| 3. 逻辑 | 4. $p \rightarrow next \quad S$ |
| 5. 栈 队列 | 6. 2 |
| 7. $n * (n+1) / 2 + 1$ | 8. 5 3 |
| 9. $2^k - 1$ | 10. 高度或深度 |
| 11. 递增 | 12. 零 |
| 13. $O(n)$ | 14. 3 |
| 15. $O(n)$ $O(\log_2 n)$ | |
| 16. 索引顺序存取方法 | 静态 |

三、名词解释

1. 指节点数据本身所占的存储量和整个节点结构所占的存储量之比。
2. 当栈满的时候, 不能进栈, 否则将产生空间溢出, 简称“上溢”。上溢是一种出错状态, 应该设法避免。
3. 所有非零元素均集中在以主对角线为中心的带状区域的矩阵。
4. 在一棵二叉树中, 除最底层外, 其余层都是满的, 并且最底层或者满的或者在右边缺少连续若干个节点。
5. 图 G 由两个集合 V 和 E 组成, 记为 $G = (V, E)$, 其中 V 是顶点 (节点) 的有穷非空集合, E 是由 V 中顶点的序偶组成的有穷集, 这些序偶称为边。

四、简答题

1. 在用头指针表示的单循环链表中, 找开始节点 a_1 的时间是 $O(1)$, 然而要找到终端节点。则需从头指针开始遍历整个链表, 其时间是 $O(n)$ 。在很多实际问题中, 表的操作常常是在表的首尾位置上进行, 此时头指针表示的单循环链表就显得不够方便。如果改用尾指针来表示单循环链表, 则查找开始节点 a_1 终端节点 a_n 都很方便, 查找时间都是 $O(1)$ 。
2. 队列的基本运算用以下六种:

Initstake (S)

构造一个空栈 S

StakeEmpty (S)

判栈空。若 S 为空栈，则返回真值，否则返回假值。

stakeFull (S)

判栈满。若 S 为满栈，则返回真值，否则返回假值。此操作只适用于栈的顺序存储结构。

Push (S, x)

进栈。若栈 S 非满，则将元素 X 插入 S 的栈顶。

Pop (S)

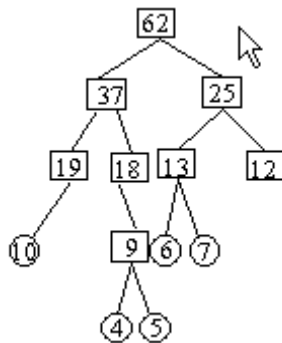
退栈。若栈 S 非满，则删去 S 的栈顶元素，并返回该元素。

StakeTop (S)

取栈顶元素。若栈 S 非空，则返回栈顶元素，但不改变栈的状态。

3. 答：链式存储在串的插入和删除操作上效率较高，因为它只需做查找工作，而不需移动后面的元素。

4.



5. 性探测再散列的散列表如下图：

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|----|---|----|----|---|---|----|---|----|---|
| 28 | 8 | 72 | 16 | 7 | 5 | 13 | 9 | 34 | |
| 1 | 1 | 1 | 2 | 5 | 1 | 1 | 6 | 3 | |

查找成功的平均查找长度为 $ASL = 119 (1 * 5 + 2 + 3 + 5 + 6) = 2.33$

五、应用题

1. converts()

```
{
    char s;
    setstack(stack);
    scanf("%c",&s),
    while (s!='*')
    {
        switch(s)
        {
            case '(': { push(stack,s); scanf("%c",&s);break; }
            case ')' : { while ((top(stack)!='') && !empty(stack))
                        { ch=pop(stack); printf("%c",ch); }
                        if(empty(stack)) { printf("%c",ch); }
                        else ch=pop(stack);
                        scanf("%c",&s); break;
                        }
            default:if(!empty(stack)) {push(stack); scanf("%c",&s);}
        }
        if(!empty( stack)) printf("no match");
    }
}
```

2. int twochild(bt tree)

```
{
    int num1,num2;
```


数据结构模拟试题（七）

第一部分 选择题

B. 中序线索二叉树中求中序后继

- C. 中序线索二叉树中求中序前趋
D. 后序线索二叉树中求后序后继
13. 某二叉树的后序遍历序列为 DABEC, 中序遍历序列为 DEBAC, 则前序序列遍历为【 】
A. ACBED B. DECAB C. DEABC D. CEDBA
14. 采用邻接表存储的图的深度优先遍历类似于二叉树的【 】
A. 先序遍历 B. 中序遍历
C. 后序遍历 D. 层次遍历
15. 快速排序在_____情况下最易发挥其长处【 】
A. 被排序的数据中含有多个相同排序码
B. 被排序的数据已基本有序
C. 被排序的数据完全无序
D. 被排序的数据中的最大值和最小值相差悬殊
16. 下述几种排序方法中, 平均查找长度最小的是【 】
A. 插入排序 B. 快速排序
C. 归并排序 D. 选择排序
17. m 阶 B-树中所有非终端(除根之外)节点中的关键字个数必须大于或等于【 】
A. $\lceil m/2 \rceil - 1$ B. $\lceil m/2 \rceil + 1$
C. $\lceil m/2 \rceil - 1$ D. m
18. 采用 Th 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. $O(n^2)$ B. $O(n \log_2 n)$
C. $O(n)$ D. $O(\log_2 n)$
19. VSAM 是虚拟存储存取方法, 该方法适于_____存取。【 】
A. 磁盘或磁带 B. 磁盘
C. 光盘 D. 磁带
20. _____的索引文件称为索引顺序文件。【 】
A. 顺序存储 B. 主文件有序
C. 磁带文件 D. 索引表有序

第二部分 非选择题

二、填空题(本大题共 16 小题, 每空 1 分, 共 20 分)。

- 一个算法的时间复杂度是指该算法的_____, 它是该算法求解问题_____的函数。
- 在线性结构中, 最后一个节点_____后续节点, 其余每个节点有且只有_____个后续节点。
- 在有 n 个节点的顺序表中做插入、删除运算, 平均时间复杂度为_____。
- 在非空队列里, 头指针始终指向_____, 尾指针始终指向_____。
- 串是由_____组成的有限序列。
- 所有模式匹配不成功的起始位置称为_____。
- 对称的 n 阶矩阵的下三角各元素存储在一维数组 V 中, 则 V 包含_____个元素。
- 广义表 $(())$ 的长度是_____。
- 一个节点的子树的个数称为这个节点的_____。
- 对于二叉树来说, 第 i 层上最多有_____个节点。
- 若 m 叉树中 A 有 3 个兄弟, 节点 B 是 A 的双亲, 则节点 B 的度是_____。
- 设有一稀疏图 G , 则 G 采用_____存储较省空间。
- n 个顶点 e 条边的图采用邻接矩阵存储, 广度优先遍历算法的时间复杂度为_____。
- 在堆排序、快速排序和归并排序中, 若只从存储空间考虑, 则应首先选取_____方法, 其次选取快速排序方法。
- 索引文件由_____和_____两部分组成。
- 在各种查找方法中, 平均查找长度与节点个数 n 无关的查找方法是_____。

三、名词解释(本大题共 5 小题, 每题 3 分, 共 15 分)

- 循环链表
- 栈
- 稀疏矩阵
- 前序遍历
- 连通分量

四、简答题(本大题共 5 小题, 每题 5 分, 共 25 分)

- 简述队列的基本运算。
- 试解释广义表 $()$ 和 $(())$ 的区别。
- 二叉树节点数值采用顺序存储结构, 如下图: 画出二叉树表示。

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| e | a | f | | d | | g | | | c | j | | | h | i | | | | | b |

4. 已知序列 [10, 18, 4, 3, 6, 12, 1, 9, 15, 8], 请给出采用希尔排序法 ($d_1=5$) 对该序列做升序排序时的每一趟的结果。

5. 设数据序列为 $D = \{13, 28, 72, 5, 16, 8, 7, 9, 34\}$, 请为 D 组织散列表。

散列函数为 $H(K) = K \% 7$, 散列表的长度为 10 个单元, 起始地址为 0, 要求用拉链法解决冲突, 并计算查找成功的平均查找长度。

五、应用题 (本大题共 2 小题, 每题 10 分, 共 20 分)

1. 有一个单链表, 其头指针为 head, 编写一个函数计算数据域为 x 的节点的个数。

2. 设计一双向冒泡排序。

数据结构模拟试题 (七) 参考答案

一、单项选择题

1. D 2. A 3. A 4. C 5. C
6. A 7. B 8. B 9. B 10. B
11. B 12. D 13. D 14. D 15. C
16. B 17. C 18. D 19. B 20. B

二、填空题

1. 时间耗费 规模 n 2. 没有 1
3. $O(n)$ 4. 队头元素 队尾元素的下一位置
5. 零个或多个字符 6. 无效位移
7. $n*(n+1)/2$ 8. 1
9. 度 10. 2^{i-1}
11. 4 12. 邻接表
13. $O(n^2)$ 14. 堆排序
15. 索引表 主文件 16. 散列查找

三、名词解释

1. 是一种首尾相连的链表。单循环链表形成一个 Next 环。而双循环链表形成 next 链环和 Prior 链环。
2. 是限制仅在表的一端进行插入和删除运算的线性表, 通常称插入、删除的这一端为栈顶, 另一端为栈底。当表中无元素时称为空栈。
3. 非零元素远远少于矩阵元素的矩阵。
4. 先访问根节点, 再前序遍历左子树, 最后前序遍历右子树, 简记为 NLR。
5. 无向图 G 中的极大连通子图称为图 G 的连通分量。

四、简答题

1. 队列的基本运算用以下六种:

InitQuenen (Q)

置空队。构造一个空队列 Q。

QuenenEmpty (Q)

判队空。若队列 Q 为空, 则返回真值, 否则返回假值。

Quenenfull (Q)

判队满。若队列 Q 为满, 则返回真值, 否则返回假值。此操作只适用于队列的顺序存储结构。

EnQuenen (Q, x)

若队列 Q 非满, 则将元素 X 插入队尾。此操作简称入队。

DeQuenen (Q)

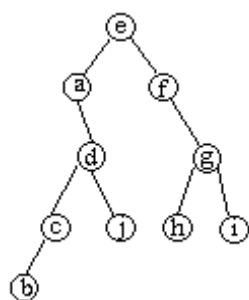
若队列 Q 为空, 则删去 Q 的队头元素, 并返回该元素。此操作简称出队。

QuenenFront (Q)

若队列 Q 非空, 则返回队头元素, 但不改变队列 Q 的状态。

2. 广义表 $()$ 是长度为 0 的空表, 对其不能做求表头和表尾的运算; 广义表 $(())$ 是长度为 1 的非空表, 对其做求表头和表尾的运算得到的表头和表尾均是空表 $()$ 。

3. 答案:

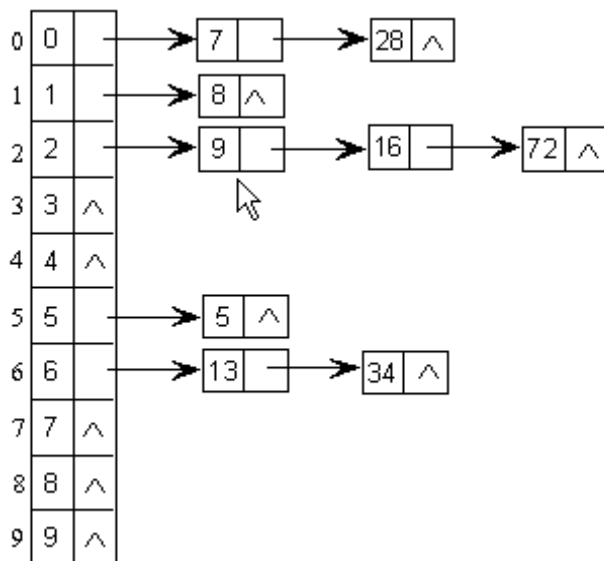


4. 第一趟 (d=5): 10, 1, 4, 3, 6, 12, 18, 9, 15,

第二趟 (d=2): 4, 1, 6, 3, 10, 8, 15, 9, 18, 12

第三趟 (d=2): 1, 3, 4, 6, 8, 9, 10, 12, 15, 18

5. 拉链法的散列表如下图:



五、应用题

- ```

int count(head)
Node *head;
{
 node *p;
 int n=0;
 p=head;
 while (p != NULL)
 {
 if (p->data == x) n++;
 p = p->next;
 }
 return(n);
}

```
- ```

void bibubble(rt r[ ],int n)
{
    int tswitch= 1,i=1,m;
    m=n- 1;
    while ((m>0) && (tswitch=1))
    {
        swich=0;
        if ((m+1) %2==0)
        {
            for(j=i;j<=m;j++)
            {
                if (r[j].key>r[j+1].key)
                {
                    tswitch=1;

```

```

        temp=r[j];
        r[j]=r[j+1];
        r[j+1]=temp
    }
    m=m-1;
}
else
{
    for(j=m;j>=i;j--)
        if (r[j+1].key<r[j].key)
        {
            tswitch= 1;temp=r[j+1];
            r[j+1]=r[j];r[j]=temp;
            i=i+1;
        }
}
}
}

```

数据结构模拟试题（八）

本试卷分两部分，第一部分为选择题，第二部分为非选择题；选择题 20 分，非选择题 80 分，满分 100 分。
考试时间 150 分钟。

第一部分 选择题

一、单项选择题（本大题共 20 小题，每小题 1 分，共 20 分）

在每小题列出的四个选项中只有一个选项是符合题目要求的，请将正确选项前的字母填在题后的括号内。

1. 数据元素是数据的基本单位，其中_____数据项【 】
A. 只能包含一个 B. 不包含
C. 可以包含多个 D. 可以包含也可以不包含
2. 下列时间复杂度中复杂度最高的是【 】
A. $O(1)$ B. $O(n)$
C. $O(\log_2 n)$ D. $O(n^2)$
3. 相对于顺序存储而言，链接存储的优点是【 】
A. 随机存取 B. 节约空间
C. 增、删操作方便 D. 节点间关系简单
4. 带头节点的单链表 head 为空的判定条件是【 】
A. head = NULL B. head -> next = NULL
C. head -> next = head D. head != NULL
5. 一个栈的入栈序列是 a, b, c, d, e, 则栈的不可能的输出序列是【 】
A. edcba B. decba
C. dceab D. abcde
6. 栈与一般线性表的区别主要在【 】
A. 元素个数 B. 元素类型
C. 逻辑结构 D. 插入、删除元素的位置
7. 串的模式匹配是指【 】
A. 判断两个串是否相等
B. 对两个串进行大小比较
C. 找某字符在串中第一次出现的位置
D. 找某子串在主串中第一次出现的位置
8. 以下说法正确的是【 】
A. 空串与空格串是相同的
B. “fox”是“Foxbase”的子串
C. 空串是零个字符的串
D. 空串长度等于 1
9. 对于二维数组 a[4][4]，数组的元素起始地址为 loc[0][0]=1000，loc[2][2] 为【 】
A. 1000 B. 1010
C. 1008 D. 1020

10. 广义表 $C = (A, B, ())$ 的表头是【 】
 A. C B. () C. A D. B
11. 具有 n 个节点的完全二叉树的深度为【 】
 A. $\lceil \log_2 n \rceil + 1$ B. $\log_2 n + 1$
 C. $\log_2 n$ D. $\lceil \log_2 n \rceil$
12. 在具有 n ($n > 1$) 个节点的完全二叉树中, 节点 i ($2i > n$) 的左孩子节点是【 】
 A. $2i$ B. $2i + 1$
 C. 不存在 D. 是 $2i - 1$
13. 对于任何一棵二叉树, 如果其终端节点数为 n , 度为 2 的节点数为 m , 则【 】
 A. $m = n + 1$ B. $n = m + 1$
 C. $n = 2m + 1$ D. $m = 2n + 1$
14. 已知一有向图的邻接表如下所示, 根据算法, 则从顶点 V_1 出发按广度优先遍历的节点序列是【 】
 A. V_1, V_2, V_3, V_4, V_5 B. V_1, V_3, V_2, V_4, V_5
 C. V_1, V_3, V_4, V_5, V_2 D. V_1, V_4, V_3, V_5, V_2
15. 堆排序是一种_____排序【 】
 A. 插入 B. 选择
 C. 交换 D. 归并
16. 快速排序的方法要求被排序的数据_____存储【 】
 A. 必须是顺序 B. 必须是链表
 C. 顺序或链表 D. 二叉树
17. 有一个有序表为 $\{1, 3, 9, 12, 32, 41, 45, 62, 75, 77, 82, 95, 100\}$ 当二分查找值为 82 的节点时, _____次比较后查找成功【 】
 A. 1 B. 2 C. 4 D. 8
18. 顺序查找法适合于存储结构为_____的线性表【 】
 A. 散列存储 B. 顺序存储或链接存储
 C. 压缩存储 D. 索引存储
19. 在顺序文件中, 所有逻辑记录在存储介质中的实际顺序与它们进入存储器的顺序【 】
 A. 一致 B. 不必一致
 C. 无关 D. 不能一致
20. 存放在外存中的数据的组织结构是【 】
 A. 数组 B. 表
 C. 文件 D. 链表

第二部分 非选择题

二、填空题(本大题共 15 小题, 每空 1 分, 共 20 分)

- 算法的时间复杂度不仅仅依赖于问题的_____，也取决于输入实例的_____。
- 选择合适的_____是解决应用问题的关键步骤。
- 链式存储方式中, 存储每个节点需要两个域, 一是_____, 二是_____。
- 当程序中同时使用两个栈时, 可以将两个栈的_____设在向量空间的两端, 仅当两个_____相遇时, 才会发生上溢。
- 稀疏矩阵的三元组中, 第 3 列存储的是稀疏数组中的_____。
- 设 $S = \text{"I AM A TEACHER"}$, 其长度等于_____。
- 在串的运算中, $\text{strcmp}(\text{"aaa"}, \text{"aabb"})$ 的值_____。
- 将一棵树转换成一棵二叉树后, 二叉树根节点没有_____子树。
- 对于一棵含有 n 个节点的完全二叉树, 它的高度是_____。

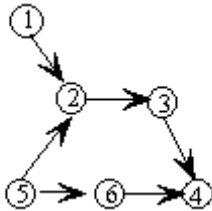
10. 三个节点可以组成_____中不同形态的树。
11. n 个节点的连通图至少有_____条边。
12. n 个顶点 e 条边的图若采用邻接矩阵存储, 则空间复杂度为_____。
13. 堆排序的时间复杂度为_____。
14. 对 n 个不同的排序码进行冒泡排序, 在元素无序的情况下比较的次数为一。
15. 常用的处理冲突的两类方法是_____和_____。
16. 文件的操作主要有两类: _____和维护。

三、名词解释 (本大题共 5 小题, 每题 3 分, 共 15 分)

1. 线性表
2. 循环队列
3. 对称矩阵
4. 中序遍历
5. 稠密图

四、简答题 (本大题共 5 小题, 每题 5 分, 共 25 分)

1. 简述头指针、头节点、开始节点的区别。
2. 比较栈和队列的异同点。
3. 简述算法复杂度的评价方法。
4. 已知某二叉树的前根排序序列是 $abedc$, 中根排序序列是 $ebdac$, 根据这两个序列能否惟一确定一棵二叉树, 若能, 请画出。
5. 写出右图的所有拓扑序列。



五、应用题 (本大题共 2 小题, 每题 10 分, 共 20 分)

1. 输出二叉树中值为 x 节点的父节点。
2. 设计实现 SHELL 排序的函数。

数据结构模拟试题 (八) 参考答案

一、单项选择题

- | | | | | |
|-------|-------|-------|-------|-------|
| 1. C | 2. D | 3. C | 4. D | 5. C |
| 6. D | 7. D | 8. C | 9. D | 10. C |
| 11. A | 12. C | 13. B | 14. B | 15. C |
| 16. A | 17. C | 18. B | 19. C | 20. C |

二、填空题

- | | |
|-----------------------------------|----------------|
| 1. 规模 初始状态 | 2. 数据结构 |
| 3. 指针域 数据域 | 4. 栈底 栈顶 |
| 5. 非零元素 | 6. 14 |
| 7. <0 | 8. 右 |
| 9. $\lfloor \log_2 n \rfloor + 1$ | 10. 2 |
| 11. $n-1$ | 12. $O(n^2)$ |
| 13. $O(n \log n)$ | 14. $n(n-1)/2$ |
| 15. 开放地址法 拉链法 | 16. 检索 |

三、名词解释

1. 是由 n ($n \geq 0$) 个数据元素 (节点) a_1, a_2, \dots, a_n 组成的有限序列。它是一种线性结构。
2. 为克服顺序队列中“假上溢”现象, 将向量空间想象为首尾相接的圆环, 存储在其中的队列称为循环队列。
3. 在一个 A 阶方阵 A 中, 若元素满足 $a[i][j] = a[j][i]$ ($0 \leq i, j \leq n-1$), 则称 A 为对称矩阵。
4. 中序遍历左子树, 再访问根节点, 最后中序遍历右子树。简记为 LNR。
5. 在很多边的图中, 对无向图边接近 $(n-1)/2$, 对有向图边接近于 $n(n-1)$, 此类图称为稠密图。

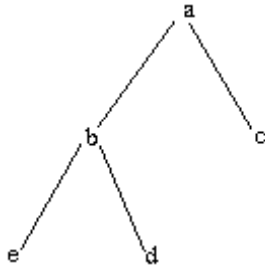
四、简答题

1. 头指针是指向链表表头节点的指针, 只要链表存在, 该指针始终不会改变。单链表由头指针惟一确定, 因此单链表可以用头指针的名字来命名。链表的头节点是在链表的开始节点之前附加的一个节点, 是链表的表头, 当链表不空时, 其内的指针指向链表的第一个节点, 当链表是空链表时, 该指针为空指针。开始节点是链表的第一个节点, 存放线性表的第一个元素。

2. 栈和队列都是加了限制的线性表，栈是先进后出表，队列是先进先出表。栈和队列的插入和删除操作都在端点进行，栈的插入和删除在同一段点，队列的插入和删除在不同的端点进行。

3. 算法复杂度可分为时间复杂度和空间复杂度。时间复杂度是该算法所耗费的时间，它是问题规模 n 的函数，可用算法的渐近时间复杂度作为时间复杂度。时间复杂度是该算法所耗费的存储，也是问题规模 n 的函数，同样可用算法的渐近空间复杂度作为空间复杂度。

4. 答案：能。



5. 答案：有两个拓扑序列：

1, 5, 2, 3, 6, 4

1, 5, 6, 2, 3, 4

五、应用题

```

1. void fitheroutput(bt *t, int x)
{
    struct node
    {
        bt *p,
        int tag;
    } s[100];
    int top=0, 1,
    while(1)
    {
        while (t!=null && t->data==x)
        {
            s[top]++,
            s[top].p=t,
            s[top].tag=0;
            t=t->left;
        }
        if (t!=null && t->data==x)
        {
            printf("father node is %d", x),
            for(i= 1;i<=top;i++) printf("%d", s[i].p->data); break,
        }
        else
        {
            while(top>0 && s[top].tag== 1 ) top- -;
            if (top>0) s[top].tag= i;
            t=s[top].p->right;
        }
    }
}

2. sortshell(ftype r, int n)
{
    int d=n, f;
    while(d>1)
    d=d/2;
    do
    {
        r=1;
        for(i= 1;i<=n-d;i++)
        {

```



```

        j=i+d;
        if (r[i].key>r[j].key)
        {
            x=r[i];
            r[i]=r[j];
            r[j]=x;
        }
    }
}
while(!f);
for (i= 1;i<=n;i++)
    printf("%d\n",r[i].key);
}

```

数据结构模拟试题（九）

第一部分选择题

一、单项选择题

1. 每一个节点只存储一个数据元素，存储节点存放在连续的存储空间，该存储方式是【】

- A. 顺序存储
- B. 链式存储
- C. 索引存储
- D. 散列存储

2. 下列算法的时间复杂度是【】

```
for (i=0; i<N; i++)
```

```
for (j=0; j<n; j++)
```

```
c[i][j]=i+j;
```

- A. $O(1)$
- B. $O(n)$
- C. $O(\log_2 n)$
- D. $O(n^2)$

3. 不带头节点的单链表 head 为空的判定条件是【】

- A. head=NULL

B. $\text{head} \rightarrow \text{next} = \text{NULL}$

C. $\text{head} \rightarrow \text{next} = \text{head}$

D. $\text{head} \neq \text{NULL}$

4. 线性表若采用链式存储结构时, 要求内存中可用存储单元的地址 【】

A. 必须是连续的

B. 部分地址必须是连续的

C. 一定是不连续的

D. 连续或不连续都可以

5. 一个栈的入栈序列是 a, b, c, 则栈的不可能的输出序列是 【】

A. acb

B. bac

C. bca

D. cab

6. 从一个栈顶指针 top 的链栈中删除一个节点时, 用 X 保存被删除的节点, 执行 【】

A. $X = \text{top}; \text{top} = \text{top} \rightarrow \text{next};$

B. $X = \text{top} \rightarrow \text{data};$

C. $\text{top} = \text{top} \rightarrow \text{next}; \quad X = \text{top} \rightarrow \text{data};$

D. $X = \text{top} \rightarrow \text{data}; \quad \text{top} = \text{top} \rightarrow \text{next};$

7. 设 $S3 = \text{"IAM"}$, $S4 = \text{"A TERCHER"}$, 则 $\text{StrCmp}(S3, S4) =$ 【】

A. "IAM"

B. "IAM A TERCHER"

C. "IAMATERCHER"

D. "ATERCHER"

8. 设二维数组 $A[0..m-1][0..n-1]$ 按行优先顺序存储, 则元素 $A[i][j]$ 的地址为 【】

A. $LOC(A[0][0]) + (j*m + i)$

B. $LOC(A[0][0]) + (j*n + i)$

C. $LOC(A[0][0]) + [(j-1)*n + i - 1]$

D. $LOC(A[0][0]) + [j-1]*m + i - 1]$

9. 广义表 $((a))$ 的表头是 【】

A. a B. (a) C. $()$ D. $((a))$

10. 广义表 $((a))$ 的表尾是 【】

A. a B. (a) C. $()$ D. $((a))$

11. 在具有 n 个节点的完全二叉树中，节点 i ($i=1$) 的父节点是 【】

A. $2i$ B. 不存在 C. $2i+1$ D. $[i/2]$

12. 若由树转化得到的二叉树是非空的二叉树，则二叉树形状 【】

A. 根节点无右子树的二叉树

B. 根节点无左子树的二叉树

C. 根节点可能有左二叉和右二叉树

D. 每个节点只有一个儿子的二叉树

13. 满二叉树_____二叉树。 【】

A. 一定是完全 B. 不一定是完全 C. 不是 D. 不是完全

14. 对于一个具有 n 个节点的无向图，若采用邻接矩阵表示，则该矩阵的大小是 【】

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

15. 有一个长度为 12 的有序表，按二分查找法对该表进行查找，在表内各元素等概率情况下，查找成功所需的平均比较次数为 【】

A. $35/12$ B. $37/12$ C. $39/12$ D. $43/12$

16. 在下列排序方法中，_____是不稳定的排序方法。 【】

A. 直接插入排序 B. 直接选择排序

C. 冒泡排序 D. 归并排序

17. 采用顺序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. n B. $n/2$ C. $(n+1)/2$ D. $(n-1)/2$

18. 如果要求一个线性表既能较快地查找, 又能适应动态变化的要求, 则可采用【】查找方法。【】

A. 顺序 B. 折半 C. 分块 D. 基于属性

19. 磁盘存储器只适应于_____存储。【】

A. 顺序 B. 随机 C. 顺序和随机 D. 散列

20. 冒泡排序的方法要求被排序的数据. 存储。【】

A. 必须是顺序 B. 必须是链表 C. 顺序或链表 D. 二叉树

第二部分非选择题

二、填空题

1. 一般情况下, 我们将算法求解问题的输入量称为问题的_____。

2. 平均时间复杂度是指所有可能的输入实例均以_____出现的情况下, 算法的期望运行时间。

3. 在线性结构中, 第一个节点_____前趋节点, 其余每个节点有且只有_____个前趋节点。

4. 一个链队列由一个_____和一个_____惟一地确定。

5. 广义表 $((a)), (b), j, ((d))$, 的长度是_____, 深度是_____。

6. 若 n 为主串长, m 为子串长, 则串的朴素匹配算法最坏的情况下需要比较字符的总次数为_____。

7. 在串的运算中, $\text{strlen}(\text{strcat}("aa", "bb"))$ 的返回值为_____。

8. 森林的后根遍历序列正是相应二叉树的_____遍历序列, 森林的先根遍历正是相应二叉树的_____遍历序列。

9. 将 5 棵树的森林转换成一棵二叉树, 则第 2 棵树的根节点成了该二叉树根节点的_____。

10. 若要求一个稀疏图 G 的最小生成树, 最好用_____算法来求解。

11. n 个顶点 e 条边的图采用邻接表存储, 深度优先遍历算法的时间复杂度为_____。

12. 在插入和选择排序中, 若初始数据基本正序, 则选用_____排序。
13. 在堆排序、快速排序和归并排序中, 若只从最坏情况下最快且节省内存考虑, 则应选取_____方法。
14. 索引文件由_____和_____两部分组成。
15. 同时发生冲突的两个关键字称为该散列函数的_____。

三. 名词解释

1. 链表
2. 递归
3. 字符串
4. 满二叉树
5. 有向图

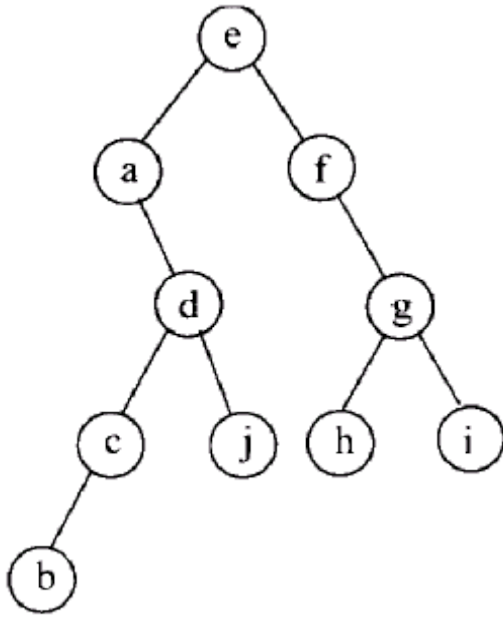
四、简答题

1. 何时选用顺序表, 何时选用链表作为线性表的存储结构为宜。
2. 简述队列的逻辑特点, 队列与线性表的异同。
3. 写出稀疏矩阵 M 对应的三元组线性表, 并画出稀疏矩阵的顺序存储结构图。

稀疏矩阵

$$M = \begin{bmatrix} 3 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 & 0 \\ 1 & 0 & 4 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}.$$

4. 将一棵二叉树 (如下图) 转化成相应森林。



5. 简述顺序查找法与二分查找法的区别。

五、应用题

- 1、设有 xtree 和 ytree 两个二叉树，xtree 中有节点 X, ytree 中有节点 Y（叶节点），将节点 X 及其子树移动到 Y 节点下。
- 2、回文是指正读和反读均相同的字符序列，如“abba”和“abdba”等均是回文。试写一个算法判定给定字符向量是否为回文。

数据结构模拟试题（九）参考答案

一、单项选择题

1. A 2. D 3. A 4. D 5. D 6. D 7. D 8. A 9. B 10. C
11. B 12. A 13. A 14. D 15. B 16. B 17. C 18. C 19. C 20. B

二、填空题

1. 规模
2. 相等概率
3. 没有 1

4. 头指针尾指针

5. 3 4

6. $(n-m+1)*m$

7. 4

8. 中序 前序

9. 右孩子

10. Kruskal

11. $O(n + e)$

12. 插入

13. 堆排序

14. 索引表 主文件

15. 同义词

三、名词解释

1. 用一组任意的存储单元来存放线性表的节点。这组存储单元既可以是连续的，也可以是不连续的。链表中节点的逻辑次序和物理次序不一定相同。

2. 若在一个函数、过程或者数据结构定义的内部直接（或间接）出现有定义本身的应用，则称它们是递归的，或者是递归定义的。

3. 由零个或多个字符组成的有限序列。

4. 一棵深度为 k 且有 $2^k - 1$ 个节点的二叉树称为满二叉树。

5. 图 G 中的每条边都是有方向的即 $E(G)$ 为有向边的集合，称该图为有向图。

四、简答题

1. 当线性表的操作主要是进行查找，很少做插入和删除操作；线性表的长度变化不大，易于事先确定其大小时，易采用顺序表作为存储结构。当线性表的长度变化较大，难以估计其存储规模，另外对线性表频繁进行插入和删除操作，易采用链表作为存储结构。

2. 队列是先进先出表。队列的插入只能在队列的一端进行，该端称为队尾。队列的删除只能从另一端进行，该端称为队头。一般线性表可以在线性表的中间及两端进行插入、删除操作。

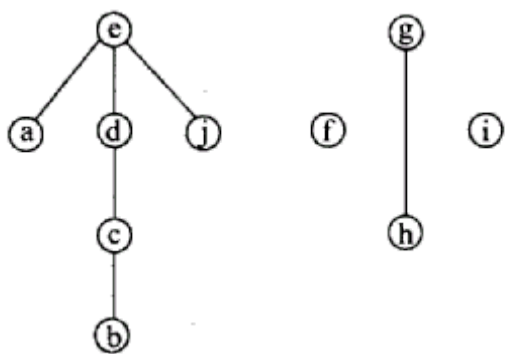
3. 答案：三元线性表：

$((1, 1, 3), (1, 4, 5), (2, 3, -2), (3, 1, 1), (3, 3, 4), (3, 5, 6), (5, 3, -1))$

顺序存储结构如下：

| | 1 | 2 | 3 |
|---|---|---|----|
| 0 | 5 | 6 | 7 |
| 1 | 1 | 1 | 3 |
| 2 | 1 | 4 | 5 |
| 3 | 2 | 3 | -2 |
| 4 | 3 | 1 | 1 |
| 5 | 3 | 3 | 4 |
| 6 | 3 | 5 | 6 |
| 7 | 5 | 3 | -1 |

4. 答案：



5. 区别：

(1) 顺序查找法与二分查找法都是在线性存储方式下进行的，顺序查找法的存储可以是顺序存储也可以是链式存储，但二分查找法必须是顺序存储。

(2) 顺序查找法对被查找的数据不要求有序；但二分查找法要求被查找的数据有序。

(3) 顺序查找法是从查找表的一端逐个查找，直到找到或查找失败；但二分查找法从被查找的数据范围的中间位置进行比较，每次排除一半数据，从另一半查找数据中查找，直到找到或查找失败。

五、应用题

1. bt *seektree(xtree,x)

```
bt **xtree
int x;
{
    bt *p, *find;
    if ((*xtree)!=null)
        { if ((*xtree)->data==x)
            { p=*xtree; *xtree=null; return(p); }
          else { find=seektree(&((*xtree)->left),x)
                if (!find) find=seektree(&((*xtree)->right),x);
                return(find);
            }
        }
    else
        return(null);
}
```

void inserttree(bt ytree,int y)

```
{
    bt *p;
    if ((*ytree)!=null)
        { if ((*ytree)->data==y)
            (*ytree)->left=seektree(xtree,x);
          else
            inserttree(&((*ytree)->left),y);
        }
}
```

2. #define StackSize 100

```
typedef char DataType;
typedef struct{
    DataType data[StackSize];
    Int top;
```

```

} SeqStack;
int example(char a[ ])
{ SeqStack *S;
    int n,i,mid;
    InitStack(S)
    n = strlen(a)
    for (i=0;i < n/2;i++) push( S,a[i]);
    i = n/2;
    while ( i<n && a[i] == a[n-i-1]) i++;
    if( i >= n ) return 1;
    else return 0;
}

```

2008-2009 年北京大学数据结构期末考试试题

(答案写在答题纸上)

一、简答（10 分）

(1) 什么是数据结构？它主要研究的问题是什么？

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。

数据结构研究的问题是数据的物理结构和逻辑结构，以及在此结构上定义并实现相关的运算，计算算法的效率。

(2) 阅读以下程序，写出带@ 语句的频度（运行次数）。

```

X=91; Y=100;
While(Y>=0)
    If (X>100) .....@
        {X -=10;Y- -}
    Else
        X++;

```

答： 频度为 1111 次

二、编程（10 分）

编制一个程序就地实现单链表的逆置算法。

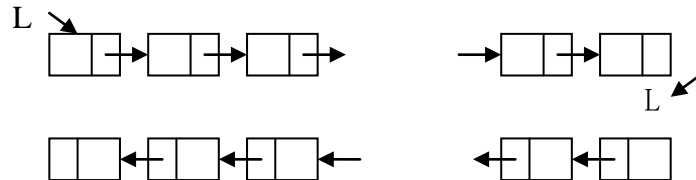
说明：1.算法中不能再另外增加结点空间(即不能在算法中

使用

malloc()函数分配新的结点空间), 但可以增加几个必要的指针变量。

2.该单链表具有头结点。

例如:



```
invert(Linklist &L)
{
    p=L;
    q=L->next;
    while(q)
    {
        r=q->next;
        q->next=p;

        p=q;
        q=r;}
    L->next=NULL;
    L=p;}

```

评分标准: 总分为 10 分。程序不同, 但思路正确者也可酌情给分。

三、编程 (10 分)

用循环队列将序列 $a_0 a_1 a_2 a_3 \dots a_{k-1} a_k \dots a_n$ 循环右移 k 位。
(序列变为 $a_k \dots a_n a_0 a_1 a_2 a_3 \dots a_{k-1}$)。

答案:

```
Right(SqQueue &Q, int k)
{
    for(I=1; I<=k; I++)
    {
        Q.base[Q.rear]=Q.base[Q.front];
        Q.rear=(Q.rear+1)%Maxsize;
        Q.front=(Q.front+1)%Maxsize;}
}

```

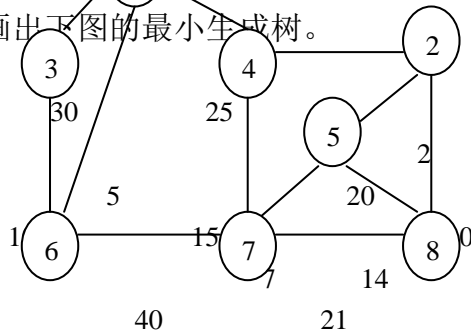
评分标准: 总分为 10 分。出队为 8 分, 入队为 2 分。程序不同, 但思路正确也可酌情给分。

四、树 (10 分)

已知一棵二叉树的先序序列为 ABDHECFGI, 中序序列为 DHBEAFCIG, 画出该二叉树。

五、画图 (1 分)

画出下图的最小生成树。



六、哈希表 (10 分)

设有一组关键字，其出现次序为：105，97，28，52，37，22，16，90，45，79，59，76，要求用哈希表方法将它们存入长度为 15 个位置的表中。

(1).构造一个哈希函数，要求发生的冲突尽可能的少(要求使用除留余数法)；

(2).用二次探测再散列法解决冲突。

写出上述各关键字的位置。

七、排序 (10 分)

给出一组关键字：66，30，78，53，6，18，10，11，20，49，当要求按升序

排序时，写出希尔排序 (d=5,3,1) 的每一趟排序结果。

八、填空 (10 分)

在 P 指针指向的结点之前插入新结点 (S 指针指向新结点)。

(提示：先将 S 结点插入 P 结点之后，然后将 S 结点和 P 结点的数据交换)

s->next=___ p->next ___;

p->next=___s___;

temp=___ s->data ___;

s->data=___ p->data ___;

p->data=___ temp___;