**第1章 绪论**

**一、选择题**

1.下列关于数据的逻辑结构的叙述中，不正确的是（C ）。

A. 数据的逻辑结构是数据间关系的描述

B. 数据的逻辑结构抽象反映数据元素间的逻辑关系

C. 数据的逻辑结构具体反映数据在计算机中的存储方式

D. 数据的逻辑结构分为线性结构和非线性结构

2. 以下关于数据的存储结构的叙述中，正确的是（ B ）。

A. 数据的存储结构是数据间关系的抽象描述

B. 数据的存储结构是逻辑结构在计算机存储器中的实现

C. 数据的存储结构分为线性结构和非线性结构

D. 数据的存储结构对数据运算的具体实现没有影响

3.假设四个算法的执行效率分别是以下四个选项，其中N是处理元素的个数，这四个算法的时间效率最高（执行时间最短）的是（ A）。

A. 1000次 B. log2N次

C. N次 D. N/100次

4.在数据结构学科中，伪代码是（ D ）。

A.描述算法且容易理解的一种语言

B.能够方便描述算法中的分支与循环等结构化语句

C.不能直接编译或解释执行

D.以上都正确

**二、判断题**

1. 数据元素是数据的最小单位。F

2. 数据项是数据不可分割的最小单位。T

3. 数据的逻辑结构是指数据的各数据项之间的逻辑关系。F

4．健壮的算法不会因非法的输入数据而出现莫名其妙的状态。T

5．算法可以用不同的语言描述，如果用C 语言或C++语言等高级语言来描述，则算法实际上就是程序了。 T

6．程序一定是算法。F

7.算法的时间复杂度仅与问题的规模相关。F

**三、填空题**

1．数据的存储结构包括 顺序 *、* 链式 *、*索引和散列存储结构。

2．数据结构中评价算法的两个重要指标是 时间复杂度 和 空间复杂度 。

3.下面程序段中带下划线的语句的执行次数的数量级是 O(log2 n) 。

i=1;

while( i<=n) i\*=2;

**四、简答题**

1.什么是信息？什么是数据? 信息和数据之间是什么关系?

信息：指音讯、消息、通信系统传输和处理对象

数据：是信息的载体，是描述客观事物的数、字符、以及所有能输入到计算机中并被计算机程序识别和处理的符号的集合

关系：在计算机中，信息必须以数据的形式出现

1. 什么是数据结构? 有关数据结构的讨论涉及哪三个方面?

数据结构：指数据以及相互之间的关系。

涉及：数据的逻辑、数据的存储、以及数据的操作

1. 数据的逻辑结构分为线性结构和非线性结构两大类。线性结构包括线性表、栈、队列、优先级队列等; 非线性结构包括树、图等。这两类结构各自的特点是什么？

线性：一对一；非线性：一对多，多对一

1. 什么是算法? 算法的5个特性是什么? 试根据这些特性解释算法与程序的区别。

算法：对特定问题的求解步骤的一种描述

特性：有穷性、确定性、可行性、输入、输出

区别：算法-有穷性-面向功能；程序-可不有穷-面向实现

5. 设*n*为正整数, 分析下列各程序段中加下划线的语句频度。n\*n\*n

for (int i = 1; i <= n; i++)

for (int j = 1; j <= n; j++) {

c[i][j] = 0.0; for (int k = 1; k <= n; k++)

c[i][j] = c[i][j] + a[i][k] \* b[k][j];

}

1. 线性表
2. ：插入删除时间复杂度的分析

双向循环链表（选择题）

**一 选择题**

1．线性表的顺序存储结构的优点是（ A ）。

A．存储密度大 B．插入运算方便

C．删除运算方便 D．可方便地用于各种逻辑结构的存储表示

2．下面关于线性表的叙述中，错误的是（ B ）。

A．线性表采用顺序存储，必须占用一片连续的存储单元

B．线性表采用顺序存储，便于进行插入和删除操作

C．线性表采用链接存储，不必占用一片连续的存储单元

D．线性表采用链接存储，便于插入和删除操作

3．线性表是具有n个（ C ）的有限序列（n≥0）。

A．数据对象 B．字符

C．数据元素 D．数据项

4．若某线性表最常用的操作是存取任意指定序号的元素和在最后进行插入和删除运算，则利用（ A ）存储方式最节省时间。

A．顺序表 B．双向链表

C．带头结点的双循环链表 D．单循环链表

5．某线性表中最常用的操作是在最后一个元素之后插入一个元素和删除第一个元素，则采用（ D ）存储方式最节省运算时间。

A．单链表 B．仅有头指针的单循环链表

C．双链表 D．仅有尾指针的单循环链表

6．设一个链表最常用的操作是在末尾插入结点和删除尾结点，则选用（ D ）最节省时间。

A. 单链表 B.单循环链表

C. 带尾指针的单循环链表 D.带头结点的双循环链表

7．若某表最常用的操作是在最后一个结点之后插入一个结点或删除最后一个结点，则采用（D ）存储方式最节省运算时间。

A．单链表 B．双链表

C．单循环链表 D．带头结点的双循环链表

//头结点可以找出最后一个结点，可快速插入.

8. 链表不具有的特点是（ B ）。

A．插入、删除不需要移动元素 B．可随机访问任一元素 //顺序表的特点

C．不必事先估计存储空间 D．所需空间与线性长度成正比

9. 若长度为n的线性表采用顺序存储结构，在其第i个位置插入一个新元素的算法的时间复杂度为（ C ）(1≤i≤n+1)。

A. O(0) B. O(1)

C. O(n) D. O(n2)

解析:顺序存储的线性表在插入新元素时，需要将第i个元素到第n个元素均向后移动一个单位，插入的新元素成为第i个元素，原来的第i个元素成为第i+1个元素，原来的第n个元素成为第n+1个元素，线性表的长度加1，插入操作的主要工作都放在移动元素上。假设线性表中含有n个数据元素，在进行插入操作时，若假定在n+1个位置上插入元素的可能性均等，则平均移动元素的个数为：除非插入在线性表的最后，否则都要移动元素。所以其时间复杂度为O(n)，答案为C。

10. 对于顺序存储的线性表，访问结点和增加、删除结点的时间复杂度为（ C ）。

A．O(n) O(n) B. O(n) O(1)

C. O(1) O(n) D. O(1) O(1)

解析:顺序存储可以实现“随机存取”，因此访问结点的时间复杂度为O(1)，而插入、删除结点由于涉及到大量移动元素，故其时间复杂度为O(n)。

11．线性表（a1,a2,…,an）以链接方式存储时，访问第i位置元素的时间复杂度为（ C ）。

A．O（i） B．O（1）

C．O（n） D．O（i-1）

12．非空的循环单链表head的尾结点p满足（ A ）。

A．p->next==head B．p->next==NULL

C．p==NULL D．p= head

13．在单链表指针为p的结点之后插入指针为s的结点，正确的操作是（ B ）。

A．p->next=s;s->next=p->next; B． s->next=p->next;p->next=s;

C．p->next=s;p->next=s->next; D． p->next=s->next;p->next=s;

14．对于一个头指针为head的带头结点的单链表，判定该表为空表的条件是（ B ）。

A．head==NULL B．head->next==NULL

C．Head->next==head D．head!=NULL

15．完成在非空双向循环链表结点p之后插入s的操作是（ D ）。

A． p->next=s ; s->prior=p; p->next->prior=s ; s->next=p->next;

B． p->next->prior=s; p->next=s; s->prior=p; s->next=p->next;

C． s->prior=p; s->next=p->next; p->next=s; p->next->prior=s ;

D． s->next=p->next; p->next->prior=s ; s->prior=p; p->next=s;

16. 在双向循环链表中，删除p所指的结点时须修改指针（ A ）。

A． p->prior->next=p->next； p->next->prior=p->prior;

B． p->prior=p->prior->prior ； p->prior->next=p;

C． p->next->prior=p； p->next=p->next->next;

D． p->next=p->prior->prior； p->prior=p->next->next;

**二、判断题**

1. 链表中的头结点仅起到标识的作用。 F

2. 顺序存储结构的主要缺点是不利于插入或删除操作。 T

3．线性表采用链表存储时，结点之间的存储空间可以是不连续的。T

4．顺序存储方式插入和删除时效率太低，因此它不如链式存储方式好。F

5. 线性表的特点是每个元素都有一个直接前驱和一个直接后继。F

6. 取线性表的第i个元素的时间同i的大小有关。 F

7. 循环链表不是线性表。 F

8. 线性表就是顺序存储的表。 F

9．为了方便地插入和删除数据，可以使用双向链表存放数据。T

10. 顺序存储方式的优点是存储密度大，且插入、删除运算效率高。 F

[解析]顺序存储方式的优点主要有：存储密度大，存储空间利用率高，便于随机存储。由于顺序存储方式在插入、删除运算会引起大量结点的移动，因此不利于经常性地执行插入和删除运算，又由于顺序存储是依靠元素的物理位置关系来反映元素之间的逻辑关系，因此对一些逻辑结构比较复杂的数据，直接存储比较困难。

**三、填空题**

1．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。

2．线性表L=（a1,a2,…,an）用数组表示，假定删除表中任一元素的概率相同，则删除一个元素平均需要移动元素的个数是 (n-1)/2 。

3．设单链表的结点结构为(data,next)，next为指针域，已知指针px指向单链表中data为x的结点，指针py指向data为y的新结点 , 若将结点y插入结点x之后，则需要执行以下语句: py->next==px->next;px->next=py; 。

4．在一个长度为n的顺序表中第i个位置（1≤i≤n+1）插入一个元素时，需向后移动 n-i+1个元素。

5．在单链表中设置头结点的作用是使插删操作变得统一，在插删第一个节点时不需要判断 。

6.顺序存储结构是通过 物理上相邻 表示元素之间的逻辑关系的;链式存储结构是通过 指针 表示元素之间的逻辑关系的。

7. 对于双向链表，在两个结点之间插入一个新结点需修改的指针共 4 个，对于单链表，在两个结点之间插入一个新结点需修改的指针共 2 个。

8. 循环单链表的最大优点是 从表中任一结点出发均可找到表中其他元素 。

9. 在单链表L中，指针p所指结点是尾结点的条件是： p->next==NULL 。

10.带头结点的双循环链表L为空表的条件是： p->next==p&&p->prior==p 。

四 应用题

1．线性表有两种存储结构：一是顺序表，二是链表。试问：

（1）如果有 n个线性表同时并存，并且在处理过程中各表的长度会动态变化，线性表的总数也会自动地改变。在此情况下，应选用哪种存储结构？ 为什么？

（2）若线性表的总数基本稳定，且很少进行插入和删除，但要求以最快的速度存取线性表中的元素，那么应采用哪种存储结构？为什么？

答：（1）选链式存储结构。它可动态申请内存空间，不受表长度（即表中元素个数）的影响插入、删除时间复杂度为O（l）

(2)选顺序存储结构。顺序表可以随机存储，时间复杂度为O（l）

2．线性表的顺序存储结构具有三个弱点：其一，在作插入或删除操作时，需移动大量元素；其二，由于难以估计，必须预先分配较大的空间，往往使存储空间不能得到充分利用；其三，表的容量难以扩充。线性表的链式存储结构是否一定都能够克服上述三个弱点，试讨论之。

答：链式存储结构一般来说克服了顺序存储结构的三个弱点

首先，插入、删除不需要移动元素，只修改指针，时间复杂度为O（l）;

其次，不需要预先分配空间，可根据需要动态申请空间；

其三，表容量只受可用内存空间的限制

其缺点是因为指针增加了空间开销，当空间不允许时，就不能克服顺序存储。

3. 设顺序表长为n，在表中插入、删除元素需要移动元素，问：

1. 在等概率情形下, 在顺序表中插入一个元素, 平均需要移动多少个元素? n/2
2. 在等概率情形下, 在顺序表进行删除一个元素, 平均需要移动多少个元素? (n-1)/2
3. 算法设计题
4. 已知线性表（a1 a2 a3 …an）按顺序存于内存，每个元素都是非零整数，试设计用最少时间把所有值为负数的元素移到全部正数值元素前边的算法：例：（x,-x,-x,x,x,-x …x）变为（-x,-x,-x…x,x,x）。

int alg2(int \*a)

{

int cnt = 0;

int low = 1, high = a[0];

while(low < high)

{

while(low < high && a[low] < 0)low++;

while(low < high && a[high] >= 0)high--;

int b = a[low];

a[low] = a[high];

a[high] = b;

cnt += 3;

}

ouf<<"cnt2="<<cnt<<":";

for(int i = 1; i <= a[0]; i++)

ouf<<a[i]<<" ";

ouf<<endl;

return cnt;

}

2. 试编写在带头结点的单链表L中删除（一个）最小值结点的（高效）算法。

3. 对线性表L=（a1...an）

（1）如L为顺序表，请设计算法将L就地逆置。

（2）若L为带头结点的单链表，设计算法将L就地逆置。

4．设有一个带头结点的单链表，其结点的数据值均为正整数，编写完成下列功能的算法：

（1）找出最小值结点，且输出该数值；

（2）若该数值是奇数，则将其与直接后继结点的数值交换；

（3）若该数值是偶数，则将其直接后继结点删除。

5.假设有两个按元素值非递减次序排列的线性表，均以单链表形式存储。请编写算法将这两个单链表归并为一个按元素值非递增次序排列的单链表，并要求利用原来两个单链表的结点存放归并后的单链表。

1. 栈和队列

第三章 ：理解栈和队列的操作；算法参考实验大纲

**一 选择题**

1. 栈对数据的操作原则是（ B ）。

A. 先进先出 B. 后进先出

C. 后进后出 D. 不分顺序

2. 一个栈的输入序列为1, 2, 3,…,n，若输出序列的第1个元素是n，则第i（1≤i≤n）个输出的元素是（ B ）。

A. 不确定 B. n-i+1

C. i D. n-i

3. 有6个元素以6，5，4，3，2，1 的顺序进栈，下列（ C ）不是合法的出栈序列。

A. 5 4 3 6 1 2 B. 4 5 3 1 2 6

C. 3 4 6 5 2 1 D. 2 3 4 1 5 6

4. 输入序列为ABC，可以变为CBA时，经过的栈操作为（ B ）。

A. push,pop,push,pop,push,pop B. push,push,push,pop,pop,pop

C. push,push,pop,pop,push,pop D. push,pop,push,push,pop,pop

5. 栈在（ D ）中应用。

A. 递归调用 B. 括号匹配的检查

C. 表达式求值 D. A，Ｂ，Ｃ

6. 一个递归算法必须包括（ B ）。

A. 递归部分 B. 终止条件和递归部分

C. 迭代部分 D.终止条件和迭代部分

7. 表达式a\*(b+c)-d的后缀表达式是（ B ）。

A．abcd\*+- B. abc+\*d-

C. abc\*+d- D. -+\*abcd

8. 设计一个判别表达式中左，右括号是否配对出现的算法，采用（ D ）数据结构最佳。

A．线性表的顺序存储结构 B. 队列

C. 线性表的链式存储结构 D. 栈

9. 用链接方式存储的队列，在进行删除运算时（ D ）。

A. 仅修改头指针 B. 仅修改尾指针

C. 头、尾指针都要修改 D. 头、尾指针可能都要修改

10. 循环队列A[0..m-1]存放其元素值，用整型变量front和rear分别表示队头和队尾指针，则当前队列中的元素个数是（ A ）。

A. (rear-front+m)%m B. rear-front+1

C. rear-front-1 D. rear-front

11. 栈和队列都是（ C ）。

A．顺序存储的线性结构 B. 链式存储的非线性结构

C. 限制存取点的线性结构 D. 限制存取点的非线性结构

12. 设栈S和队列Q的初始状态为空，元素e1,e2,e3,e4,e5和e6依次通过栈S，一个元素出栈后即进队列Q，若6个元素出队的序列是e2,e4,e3,e6,e5,e1，则栈S的容量至少应该是（ C ）。

A．6 B. 4

C. 3 D. 2

**二 判断题**

1. 消除递归不一定需要使用栈。T / /消除递归不一定得用到栈,有时候用循环或者队列也可以解决

2. 栈是实现过程和函数等子程序所必需的结构。T

3. 栈和队列都是限制存取点的线性结构。T

4. 通常使用队列来处理函数或过程的调用。F

解析:通常使用栈来处理函数或过程调用

1. 循环队列也存在空间溢出问题。T

解析:只要是一片大小固定的存储空间但不允许对已存储数据进行覆盖，只要达到空间存满时再往此存储空间内添加存储元素数据均会引起溢出.

6. 栈和队列都是线性表，只是在插入和删除时受到了一些限制。T

7. 栈和队列的存储方式，既可以是顺序方式，又可以是链式方式。T

1. **填空题**

1．栈是 限定在表尾进行插入或删除（操作受限） 的线性表，其运算遵循后进先出的原则。

2. 一个栈的输入序列是：1 2 3，则不可能的栈输出序列是 312 。

3．为了增加内存空间的利用率和减少溢出的可能性，由两个栈共享一片连续的内存空间时，应将两栈的 栈底分别设在这片内存空间的两端，这样栈满的条件是两栈顶指针值相减的绝对值为1（或两栈顶指针相邻）。

4. 多个栈共存时，最好用 链式 存储结构。

5．用S表示入栈操作，X表示出栈操作，若元素入栈的顺序为1234，为了得到1342出栈顺序，相应的S和X的操作串序列为： SXSSXSXX 。

6. 循环队列的引入，目的是为了 克服假溢出时大量元素的移动 。

7. 表达式求值是 栈 应用的一个典型例子。

8. 已知循环队列Q.base[0..m-1]，队首和队尾指针分别为Q.front和Q.rear，队列满的条件是 （Q.rear+1）%m==Q.front ，队列空的条件 Q.front== Q.rear 。

9. 在解决计算机主机与打印机之间速度不匹配问题时通常设置一个打印数据缓冲区，主机将要输出的数据依次写入该缓冲区，而打印机则从该缓冲区中依次取出数据打印，该缓冲区是一个 队列 结构。

**四 应用题**

1. 如果输入序列为1 2 3 4 5 6,试问能否通过栈结构得到以下两个序列:4 3 5 6 1 2和1 3 5 4 2 6;请说明为什么不能或如何才能得到。

S：入栈；X：出栈

435612不可得到；135426可以得到，顺序为：SXSSXSSXXXSX

1. （1） 什么是递归程序？

答：一个函数在结束本函数之前直接或间接调用函数

（2） 递归程序的优、缺点是什么？

答：优点-结构简单、清晰、易证明其正确性；缺点-执行中占内存空间较多，运行效率低 。

（3） 递归程序在执行时，应借助于什么数据结构来完成？

答：递归程序

3.什么是循环队列？

**五 算法设计题**

1. 检查表达式中括号是否匹配。

2. 利用栈和队列，判断键盘上输入的n个数是否构成回文序列。

3. 利用栈的基本操作实现将十进制整数N转换为r（2≤r≤16）进制数，并输出。

1. 串
2. ：顺序存储（定长，堆）

串的比较

BF算法（阅读，填空）

**一、选择题**

1．下面关于串的的叙述中，哪一个是不正确的？ B

A．串是字符的有限序列 B．空串是由空格构成的串

C．模式匹配是串的一种重要运算 D．串既可以采用顺序存储，也可以采用链式存储

2.串是一种特殊的线性表，其特征体现在\_\_B\_\_\_。

A. 可以顺序存储 B. 数据元素是一个字符

C. 可以链接存储 D. 数据元素可以是多个字符

3．设有两个串p和q，其中q是p的子串，求q在p中首次出现的位置的运算称为（ C ）

A．求子串 B．联接

C．模式匹配 D．求串长

4．若串S="software",其子串的数目是（ B ）。

A．8 B．37

C．36 D．9

5．串的长度是指（ B ）。

A．串中所含不同字母的个数 B．串中所含字符的个数

C．串中所含不同字符的个数 D．串中所含非空格字符的个数

6．在字符串匹配的BF算法中，当主串位i与模式串位j比较失败时，新一趟匹配开始，主串的位移公式是（ D ）。

A．i=i+1 B．i=j+1

C．i=i-j+1 D．i=i-j+2

**二、判断题**

1. 串是一种数据对象和操作都特殊的线性表。T
2. 空格串和空串的长度均为1。F
3. 判断两个串是否相等，只需要判断这两个串是否包含相同的字符即可。F
4. 如果一个串中的所有字符均在另一串中出现，那么则说明前者是后者的子串。F
5. KMP算法的最大特点是指示主串的指针不需回溯。 T
6. 设模式串的长度为m,目标串的长度为n，当n≈m且处理只匹配一次的模式时，朴素的匹配（即子串定位函数）算法所花的时间代价可能会更为节省。T

**三、填空题**

1. 串是一种特殊的线性表，其特殊性表现在 数据元素是字符，且其基本操作或串的整体或子串的部分 ；串的两种最基本的存储方式是 顺序 、 链式 。
2. 两个串相等的充分必要条件是 串长相等且字符对应相等 。组成串的数据元素只能是 字符 。
3. 一个字符串中 任意个连续的字符组成的子序列 称为该串的子串 。
4. index（"datastructure"，"str",1）=\_\_ 5 \_\_\_。
5. 设正文串长度为n，模式串长度为m，则串匹配的KMP算法的时间复杂度为 O（m+n） 。
6. 模式串"abaabcac"的next函数值序列为 *01122312* 。

**四、算法设计**

1.设计算法以判断串T是否是串S的子串，若是子串，返回1，否则返回0。

2.设计算法比较串S1和串S2的大小。（S1<S2，返回-1；S1＝S2，返回0；S1<S2，返回1。

第 5 章 数组和广义表

第五章无算法题

一、选择题

1. 若对n阶对称矩阵A以行序为主序方式将其下三角的元素(包括主对角线上所有元素)依次存放于一维数组B［1..(n(n+1))/2］中，则在B中确定aij（i≤j）的位置k的关系为( B )。

A. i\*(i-1)/2+j B. j\*(j-1)/2+i

C. i\*(i+1)/2+j D. j\*(j+1)/2+i

2. 对稀疏矩阵进行压缩存储目的是( C )。

A．便于进行矩阵运算 B．便于输入和输出

C．节省存储空间 D．降低运算的时间复杂度

3.稀疏矩阵一般的压缩方法有两种，即( C )。

A.二维数组和三维数组 B.三元组和散列

C.三元组顺序表和十字链表 D.散列和十字链表

4. 已知广义表LS＝((a,b,c),(d,e,f)),运用head和tail函数取出LS中原子e的运算是( C )。

A. head(tail(LS)) B. tail(head(LS))

C. head(tail(head(tail(LS))) D. head(tail(tail(head(LS))))

5. 广义表((a),a)的表头和表尾分别是( B )。

A. a ,((a)) B. (a) ,(a)

C. (a),a D. ((a)) , a

6. 广义表A=(a,b,(c,d),(e,(f,g)))，则Head(Tail(Head(Tail(Tail(A)))))的值为( D )。

A. (g) B. (d)

C. c D. d

7. 下面说法不正确的是( A )。

A. 广义表的表头总是一个广义表 B. 广义表的表尾总是一个广义表

C. 广义表难以用顺序存储结构表示 D. 广义表可以是一个多层次的结构

二、判断题

1. 数组可看成线性结构的一种推广，因此与线性表一样，可以对数组进行插入，删除等操作。F

2. 一个稀疏矩阵Am\*n采用三元组形式表示， 若把三元组中有关行下标与列下标的值互换，并把m和n的值互换，则就完成了Am\*n的转置运算。 F

3. 广义表的取表尾运算，其结果通常是个表，但有时也可是个单元素。F

4. 若一个广义表的表头为空表，则此广义表亦为空表。F

5. 所谓取广义表的表尾就是返回广义表中最后一个元素。F

6. 一个广义表可以为其它广义表所共享。T

三、 填空题

1. 数组采用 顺序 存储方式。

2. 所谓稀疏矩阵指的是 非零元素很少且分布没有规律 的矩阵。

3. 对特殊矩阵进行压缩的目的是 节约存储空间 。

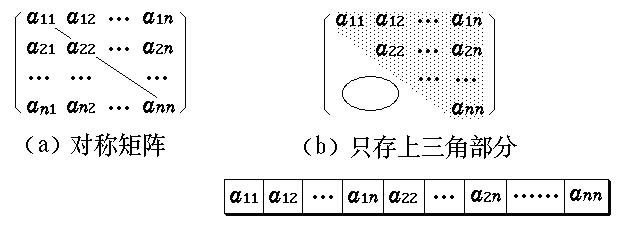
4. 当广义表中的每个元素都是原子时，广义表便成了 线性表 。

5．设广义表L=((),()), 则head(L)是 （） ；tail(L)是 （（）） ；L的长度是 2 ；深度是 2 。

四、应用题

1. 设有一个n×n的对称矩阵A，如图(a)所示。为了节约存储，可以只存对角线及对角线以上的元素，称为上三角矩阵。我们把它们按行存放于一维数组B [1..n(n+1)/2]中，如图(b)所示，并称之为对称矩阵A的压缩存储方式。

试问：对称矩阵中的任一元素aij在对应存于一维数组B的下标位置k的计算公式是什么?



思考：

1. 如果按按照列主序存储上三角的元素，则A中任一元素aij应在数组B中的下标位置k是什么？

k=[j(j-1)]/2+i ;i<=j k=[i(i-1)]/2+j ;i>j

1. 如何按照行主序（或列主序）存储下三角的元素，则A中任一元素aij应在数组B中的下标位置k是什么？

行主序 k=[i(i-1)]/2+j-1 ;i<=j k=[j(j-1)]/2+i-1 ;i>j

列主序 [(j-1)(2n-j+2)]/2+i-j+1 ;i<=j k=[(i-1)(2n-i+2)]/2+j-i+1 ;i>j

1. 特殊矩阵和稀疏矩阵哪一种压缩存储后失去随机存取的功能？为什么？

答：（1）特殊矩阵：仍具有随机存储功能

（2）稀疏矩阵:失去了随机存储

1. 一个n×n的对称矩阵，如果以行或列为主序存入内存，则其容量为多少？

答：n(n+1)/2 (压缩存储)

n\*n (不采用压缩存储)

1. 数组，广义表与线性表之间有什么样的关系？

数组:具有相同性质的数据元素的集合，每一个元素有唯一的下标限定，n维数组可以看做其元素是n-1维数组的一个线性表

广义表：元素可以是原子，也可以是子表，即广义表是原子或子表的有限序列，是线性表的推广

线性表：元素可以个种各样，但必须具有相同性质，属于同一数据对象

五、算法设计题

1.设稀疏矩阵M采用三元组顺序表进行了压缩存储，请写出稀疏矩阵M的转置算法。

2.设稀疏矩阵M采用十字链表做存储结构，请写出将其按照矩阵的行列结构输出的算法。

第6章 树和二叉树

第六章：二叉树的相关内容（二叉链表，遍历，应用（统计节点等）），什么是线索二叉树，重点。

建议熟悉该章实验大纲与上机考试的代码（个人建议）

**一、选择题**

1. 设树T的度为4，其中度为1，2，3和4的结点个数分别为4，2，1，1 则T中的叶子数为（ D ）。

A．5 B．6

C．7 D．8

2. 一棵具有 n个结点的完全二叉树的高度（深度）是（ A ）。

A．⎣log2n⎦+1 B．log2n+1

C．⎣log2n⎦ D．log2n-1

3．高度为 k的二叉树最大的结点数为（ C ）。

A．2k B．2k-1

C．2k -1 D．2k-1-1

4. 利用二叉链表存储树，则根结点的右指针（ C ）。

A．指向最左孩子 B．指向最右孩子

C．为空指针 D．为非空指针

5．森林的先序遍历序列等同于对应的二叉树的( A ).

A. 先序序列 B. 中序序列

C. 后序序列 D. 层次序列

6.已知一棵二叉树的先序遍历结果为ABCDEF,中序遍历结果为CBAEDF,则后序遍历的结果为（ A ）。

A．CBEFDA B． FEDCBA

C．CBEDFA D．不确定

7. 引入线索二叉树的目的是（ A ）

A．加快查找结点的前驱或后继的速度 B．为了能在二叉树中方便地进行插入与删除

C．为了能方便的找到双亲 D．使二叉树的遍历结果唯一

8．3 个结点的二叉树有（ D ）种不同的形态。

A．2 B．3

C．4 D．5

**二、判断题**

1. 二叉树是度为2的有序树。 F
2. 对一棵二叉树进行层次遍历时，应借助于一个栈。F
3. 树的先根遍历和其相应的二叉树的先序遍历的结果是一样的。T
4. 中序遍历一棵二叉排序树的结点就可得到排好序的结点序列。T
5. 根据任何一棵二叉树的先序序列和后序序列可以唯一确定这棵二叉树。F
6. 二叉树是树的特殊情形。F
7. 树形结构中元素之间存在一对多的关系。F
8. 完全二叉树的存储结构通常采用顺序存储结构。F
9. 树与二叉树是两种不同的树型结构。T
10. 哈夫曼树的结点个数不能是偶数。T

**三、填空题**

* 1. 二叉树由 数据域、左、右指针 三个基本单元组成。
  2. 在二叉树中，指针p所指结点为叶子结点的条件是 p->lchild==NULL&&p->rchild==NULL 。
  3. 具有N个结点的二叉树，采用二叉链表存储，共有 N+1 个空链域。
  4. 一个无序序列可以通过构造一棵 二叉排序树 树而变成一个有序序列，构造树的过程即为对无序序列进行排序的过程。
  5. 若以{4，5，6，7，8}作为叶子结点的权值构造哈夫曼树，则其带权路径长度是 69 。
  6. 设n0为哈夫曼树的叶子结点数目，则该哈夫曼树共有 2n0+1 个结点。

**四、应用题**

1．试分别画出具有3个结点的不同形态的树和二叉树。

2．对于任何一棵非空的二叉树,假设叶子结点的个数为n0,而次数为2的结点个数为n2,请给出n0和n2之间所满足的关系式。

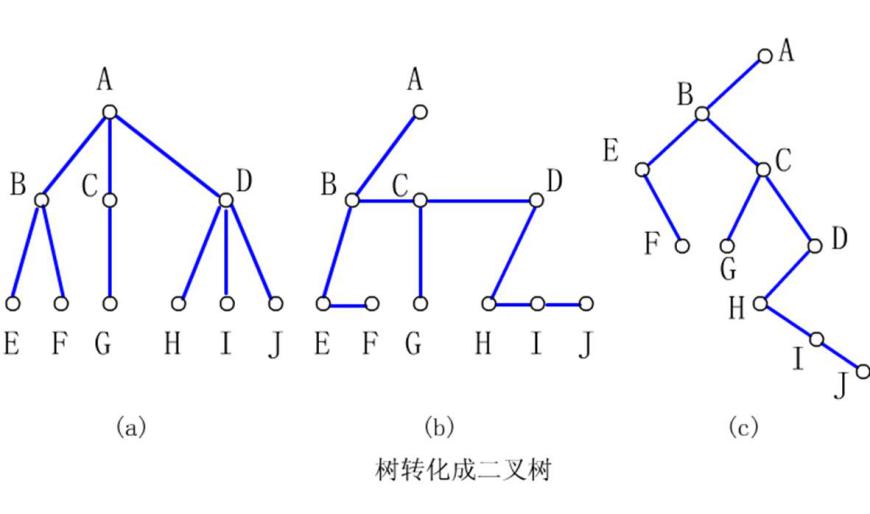
3.将算术表达式（a+b）-c\*(d+e)/f转化为二叉树，并根据该二叉树，求出其前缀和后缀表达式。

4.已知一棵二叉树的先序遍历的结果是ABECDFGHIJ, 中序遍历的结果是EBCDAFHIGJ, 试画出这棵二叉树。

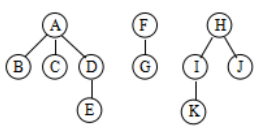
1. 设有正文AADBAACACCDACACAAD，字符集为{A,B,C,D},设计一套二进制编码，使得上述正文的编码最短。
2. 假定用于通信的电文仅由8个字母c1, c2, c3, c4, c5, c6, c7, c8组成, 各字母在电文中出现的频率分别为5, 25, 3, 6, 10, 11, 36, 4。试为这8个字母设计不等长Huffman编码, 并给出该电文的总码数。

7.二叉树、树和森林是三种不同的数据结构，问：

1. 指出树和二叉树的主要区别。
2. 将下图所示的树转化为二叉树。



1. 将下图所示的森林转化为二叉树。



1. 将树和森林转换为二叉树的基本目的是什么？

五、算法设计题（参加实验大纲）

1. 图

第七章：邻接表，邻接矩阵，遍历（深搜、广搜）{可能以简答题形式}，该章只要涉及邻接矩阵，则时间复杂度为O(n\*n),涉及邻接表，则时间复杂度为O（n+e）；最小生成树；关键路径（概念），求最短路径（应用题）

**一、选择题**

1．设无向简单图的顶点个数为n，则该图最多有（ B ）条边。

A．n-1 B．n(n-1)/2

C． n(n+1)/2 D．n2

2．一个n个顶点的连通无向图，其边的个数至少为（ A ）。

A．n-1 B．n

C．n+1 D．nlogn；

3．在一个无向图中，所有顶点的度数之和等于所有边数（ B ）倍，在一个有向图中，所有顶点的入度之和等于所有顶点出度之和的（ C ）倍。

A．1/2 B．2

C．1 D．4

4．下列（ B ）图的邻接矩阵一定是对称矩阵。

A．有向无环图 B．无向图

C．AOV网 D．AOE网

5．下列有关图的遍历的说法中，不正确的是（ C ）。

A．用邻接表存储的图的深度优先搜索的时间复杂度为O(n＋e)

B．图的广度优先搜索中邻接点的寻找具有“先进先出”的特征，需要采用队列结构来实现

C．非连通图不能用深度优先搜索法

D．图的遍历要求每一顶点访问且仅被防问一次

6. 求解最短路径的Floyd算法的时间复杂度为( D )。

A．O（n） B. O（n+c）

C. O（n2） D. O（n3）

7. 在用邻接表表示图时，拓扑排序算法时间复杂度为( B )。

A. O(n) B. O(n＋e)

C. O（n2） D. O（n3）

9. 关键路径是AOE网中（ A ）。

A．从源点到汇点的最长路径 B．从源点到汇点的最短路径

C．最长回路 D．最短回路

10．下列关于AOE网的叙述中，不正确的是（ B ）。

A．关键活动不按期完成就会影响整个工程的完成时间

B．任何一个关键活动提前完成，那么整个工程将会提前完成

C．所有的关键活动提前完成，那么整个工程将会提前完成

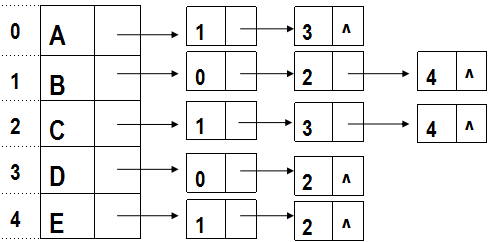
D．某些关键活动提前完成，那么整个工程将会提前完成

**二、判断题**

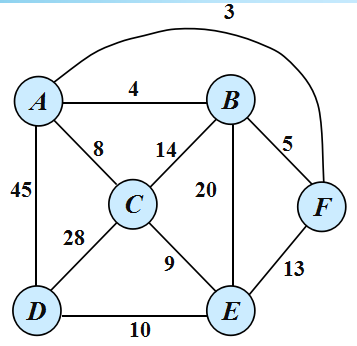
1. 有e条边的无向图，在邻接表中有e个结点。F
2. 强连通图的各顶点间均可达。T
3. 十字链表是无向图的一种存储结构。F
4. 用邻接矩阵存储一个图时，在不考虑压缩存储的情况下，所占用的存储空间大小与图中结点个数有关，而与图的边数无关。 T
5. 带权的连通无向图的最小生成树是唯一的。T
6. AOV网的含义是以边表示活动的网。F

**三、 应用题**

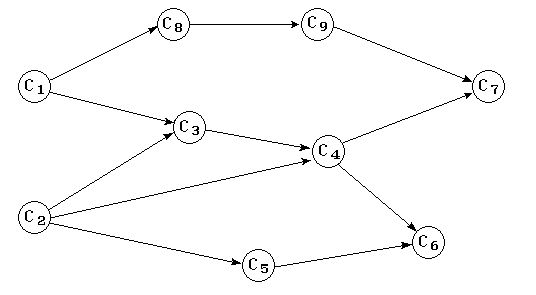
1.已知一个无向图G的邻接表存储表示如下，试写出从顶点A出发进行深度和广度优先遍历得到的顶点序列，并判断该图的连通性。



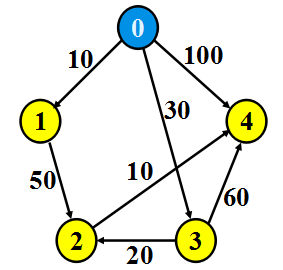
1. 下图表示一个地区的通讯网，边表示城市间的通讯线路，边上的权表示架设线路花费的代价，如何选择能沟通每个城市且总代价最省的n-1条线路，画出你的选择，并写出这n-1条路的总代价。



1. 叙述拓扑排序的基本思想，并对如下的有向图，写出两个不同的拓扑序列。



1. 写出如下有向网的邻接矩阵，并应用Dijkstra算法求出从顶点0到其余各顶点的最短路径。



**四、 算法设计题**

已知图的邻接表存储定义如下：

#define MAX\_VEX\_NUM 20 //图的最大顶点个数

typedef struct ArcNode {

int adjvex; // 该弧所指向的顶点的位置

struct ArcNode \*nextarc; // 指向下一条弧的指针

} ArcNode;

typedef struct VNode {

VertexType data; // 顶点信息

ArcNode \*firstarc; // 指向第一条依附该顶点的弧

} VNode, AdjList[MAX\_VERTEX\_NUM];

typedef struct {

AdjList vertices;

int vexnum, arcnum; // 顶点个数和弧数

int kind; // 图的种类标志

} ALGraph;

(1)试写出计算图中所有顶点的入度算法，并将每个顶点的入度存入数组indegree中。

void FindIndegree(ALGraph G,int indegree[ ])

(2)试写出计算图中所有顶点的出度算法，并将每个顶点的出度存入数组outdegree中。

void FindOutdegree(ALGraph G,int outdegree[ ])

1. 查找
2. ：顺序查找，平均查找长度，时间复杂度

折半查找，算法分析，填空

二叉排序树（重要）

哈希函数

**一、选择题**

1． 对N个元素的表做顺序查找时，若查找每个元素的概率相同，则平均查找长度为( A )。

A．(N+1)/2 B. N/2

C. N D. [(1+N)\*N ]/2

2．下面关于折半查找的叙述正确的是 ( D ) 。

A. 表必须有序，表可以顺序方式存储，也可以链表方式存储

B. 表必须有序，而且只能从小到大排列

C. 表必须有序.且表中数据必须是整型，实型或字符型

D. 表必须有序，且表只能以顺序方式存储

3．折半查找的时间复杂度为（ D ）。

A. O（n2） B. O（n）

C. O（nlogn） D. O（logn）

4．当采用分块查找时，数据的组织方式为 ( B ) 。

A．数据分成若干块，每块内数据有序

B．数据分成若干块，每块内数据不必有序，但块间必须有序，每块内最大（或最小）的数据组成索引块

C. 数据分成若干块，每块内数据有序，每块内最大（或最小）的数据组成索引块

D. 数据分成若干块，每块（除最后一块外）中数据个数需相同

5．二叉排序树的查找效率与二叉树的 （1）C 有关, 在 （2）C 时其查找效率最低。

(1) A. 高度 B. 结点的多少 C. 树型 D. 结点的位置

(2) A. 结点太多 B. 完全二叉树 C. 呈单枝树 D. 结点太复杂。

6. 设有一组记录的关键字为{19，14，23，1，68，20，84，27，55，11，10，79}，用链地址法构造散列表，散列函数为H（key）=key % 13，散列地址为1的链中有（ D ）个记录。

A．1 B. 2

C. 3 D. 4

7.设哈希表长为14，哈希函数是H(key)=key%11,表中已有数据的关键字为15，38，61，84，要将关键字为49的结点加到表中，用二次探测再散列法解决冲突，则放入的位置是( D ) 。

A．8 B．3

C．5 D．9

**二、判断题**

1. 折半查找法的查找速度一定比顺序查找法快 。 F
2. N个结点的二叉排序树有多种，其中树高最小的二叉排序树是最佳的。T
3. 在任意一棵非空二叉排序树中，删除某结点后又将其插入，则所得二排序叉树与原二排序叉树相同。 F
4. 向某个平衡因子不为零的结点的平衡二叉树树中插入一新结点，必引起平衡旋转。 F
5. 二叉排序树删除一个结点后，仍是二叉排序树。T
6. 采用线性探测法处理散列时的冲突，当从哈希表删除一个记录时，不应将这个记录的所在位置置空，因为这会影响以后的查找。T
7. 在哈希查找中，“比较”操作一般也是不可避免的。T
8. 哈希函数越复杂越好，因为这样随机性好，冲突概率小。 F

**三、 填空题**

1. 顺序查找n个元素的顺序表，若查找成功，则比较关键字的次数最多为\_\_ n \_\_次；当使用监视哨时，若查找失败，则比较关键字的次数为\_\_ n+1 \_\_。
2. 在顺序表（8,11,15,19,25,26,30,33,42,48,50）中，用二分（折半）法查找关键码值20，需做的关键码比较次数为 4 \_。
3. 分块查找中，若索引表和各块内均用顺序查找，则有900个元素的线性表分成\_ 30\_\_\_块最好。
4. 动态查找表和静态查找表的重要区别在于前者包含有 插入 和\_删除\_\_\_\_\_\_运算，而后者不包含这两种运算。
5. 哈希表是通过将查找码按选定的 哈希函数 和 解决冲突的办法 ，把结点按查找码转换为地址进行存储的线性表。哈希方法的关键是 选择好的哈希函数 \_\_和 处理冲突的办法 \_\_。一个好的哈希函数其转换地址应尽可能\_\_ 均匀 \_\_，而且函数运算应尽可能\_ 简单 \_\_。
6. 在哈希函数H（key）=key%p中，p值最好取 小于等于表长的最大素数或者不包含小于的质因子的合数 。

**四、 应用题**

1.用序列(46,88,45,39,70,58,101,10,66,34)建立一个排序二叉树，画出该树，并求在等概率情况下查找成功的平均查找长度。

ASL=32/10

2. 对下面的关键字集{30,15,21,40,25,26,36,37}若查找表的装填因子为0.8，采用线性探测再散列方法解决冲突。

（1）设计哈希函数；

（2）画出哈希表；

（3）计算查找成功和查找失败的平均查找长度。

答：由于装填因子为0.8.关键字有8个，所以表长为8/0.8=10

1. 用除留余数法，哈希函数为H(key)=key%7

(3) ASL=16/8=2

1. **算法设计题**
2. 折半查找算法
3. 二叉排序树的查找、插入算法。

第10章 内部排序

**一、选择题**

1．某内排序方法的稳定性是指（ D ）。

A．该排序算法不允许有相同的关键字记录

B．该排序算法允许有相同的关键字记录

C．平均时间为0（n log n）的排序方法

D．以上都不对

2．下列排序算法中，其中（ D ）是稳定的。

A. 堆排序，冒泡排序 B. 快速排序，堆排序

C. 直接选择排序，归并排序 D. 归并排序，冒泡排序

1. 下列内部排序算法中：

A．快速排序 B.直接插入排序

C. 二路归并排序 D. 简单选择排序

E. 起泡排序 F. 堆排序

问：

（1）比较次数与序列初态无关的算法是（ CD ） 。

（2）不稳定的排序算法是（ ADF ）。

（3）在初始序列已基本有序的情况下，排序效率最高的算法是（ B ）。

（4）排序的平均时间复杂度为O(nlogn)的算法是（ ACF ），为O(n2)的算法是（ BDE ）。

4．数据序列（8，9，10，4，5，6，20，1，2）只能是下列排序算法中的（ C ）的两趟排序后的结果。

A.选择排序 B.冒泡排序

C.插入排序 D.堆排序

5．下列排序算法中（ B ）不能保证每趟排序至少能将一个元素放到其最终的位置上。

A.快速排序 B. shell排序

C. 堆排序 D.冒泡排序

6. 一组记录的关键码为（46，79，56，38，40，84），则利用快速排序的方法，以第一个记录为基准得到的一次划分结果为（ C ）。

A．(38,40,46,56,79,84) B. (40,38,46,79,56,84)

C．(40,38,46,56,79,84) D. (40,38,46,84,56,79)

7. 下列排序算法中，在待排序数据已有序时，花费时间反而最多的是（ C ）排序。

A．冒泡 B. 希尔

C. 快速 D. 堆

8.在下面的排序方法中，辅助空间为O（n）的是（ D ） 。

A．希尔排序 B. 堆排序

C. 选择排序 D. 归并排序

9.就平均性能而言，目前最好的内排序方法是（ D ）排序法。

A. 冒泡 B. 希尔插入

C. 交换 D. 快速

10.如果只想得到1000个元素组成的序列中第5个最小元素之前的部分排序的序列，用（ D ）方法最快。

1. 起泡排序 B．快速排序
2. C．Shell排序 D．堆排序 E．简单选择排序

11.快速排序方法在（ D ）情况下最不利于发挥其长处。

A. 要排序的数据量太大 B. 要排序的数据中含有多个相同值

C. 要排序的数据个数为奇数 D. 要排序的数据已基本有序

12.下列四个序列中，哪一个是堆（ C ）。

A. 75,65,30,15,25,45,20,10 B. 75,65,45,10,30,25,20,15

C. 75,45,65,30,15,25,20,10 D. 75,45,65,10,25,30,20,15

13.有一组数据（15，9，7，8，20，-1，7，4），用堆排序的筛选方法建立的初始堆为（ C ）。

A．-1，4，8，9，20，7，15，7 B．-1，7，15，7，4，8，20，9

C．-1，4，7，8，20，15，7，9 D．A，B，C均不对。

14.就排序算法所用的辅助空间而言，堆排序，快速排序，归并排序的关系是（ A ）。

A．堆排序<快速排序<归并排序 　 B．堆排序<归并排序<快速排序

C．堆排序> 归并排序>快速排序 　 D．堆排序>快速排序>归并排序

15．从未排序序列中依次取出一个元素与已排序序列中的元素依次进行比较，然后将其放在已排序序列的合适位置，该排序方法称为（ A ）排序法。

A. 插入 B. 选择

C. 希尔 D. 二路归并

**二、判断题**

1. 排序的稳定性是指排序算法中的比较次数保持不变，且算法能够终止。F
2. 堆肯定是一棵平衡二叉树。F
3. 在初始数据表已经有序时，快速排序算法的时间复杂度为O(nlog2n )。F
4. 在用堆排序算法排序时，如果要进行增序排序，则需要采用“大根堆”。 F
5. （101，88，46，70，34，39，45，58，66，10）是堆。 T
6. 当待排序的元素很大时，为了交换元素的位置，移动元素要占用较多的时间，这是影响时间复杂度的主要因素。T
7. 折半插入排序所需比较次数与待排序记录的初始排列状态相关。F
8. 快速排序的速度在所有排序方法中为最快,而且所需附加空间也最少。F

**三、 填空题**

1. 设表中元素的初始状态是按健值递增的，分别用堆排序，快速排序，冒泡排序和归并排序方法对其进行排序（按递增顺序）， 冒泡 排序最省时间， 快速 排序最费时间。
2. 不受待排序初始序列的影响，时间复杂度为O(N2)的排序算法是 简单选择排序 ，在排序算法的最后一趟开始之前，所有元素都可能不在其最终位置上的排序算法是 直接插入排序（最小的元素在最后时） 。
3. 直接插入排序用监视哨的作用是\_ 免去查找过程中每一步都要检测整个表是否查找完毕，提高了查找效率 \_。
4. 在数据表有序时，快速排序算法的时间复杂度是 O（n\*n） \_\_。
5. 堆是一种有用的数据结构. 堆排序是一种 选择 排序，堆实质上是一棵 完全二叉树 结点的层次序列。对含有N个元素的序列进行排序时，堆排序的时间复杂度是 O（Nlog2 N） (2为对数的底) ，所需的附加存储结点是 O（l） 。

**四、 应用题**

1.内部排序（名词解释）。

答：在内存里面进行

2.在各种排序方法中，哪些是稳定的？哪些是不稳定的？并为每一种不稳定的排序方法举出一个不稳定的实例。

3.以下概念的区别：拓扑排序与冒泡排序。

答：拓扑排序--是有向图的顶点依照弧的走向，找出一个全序集的过程，主要是根据与顶点连接的弧来确定顶点序列

冒泡排序--借助交换思想通过比较相邻节点关键字大小进行排序的算法

4.对于堆排序法，快速排序法和归并排序三种方法来说

（1）若仅从节省存储空间考虑，则应该首先选取其中哪种方法？其次选取哪种方法？

（2）若仅考虑排序结果的稳定性，则应该选取哪种方法？

（3）若仅从平均情况下排序最快这一点考虑，则应该选取哪种方法？

答：（1）先选堆排序，再选快排，最后选择归并排序

（2）选择归并。堆排序和快速排序都是不稳定排序

（3）快排

1. 快速排序的最大递归深度是多少？最小递归深度是多少？

答：设待排序记录的个数为n，则快排的最小递归深度为[log2 n]+1,最大递归深度为n。

6.请回答下列关于堆（Heap）的一些问题：

（1）堆的存储表示是顺序的，还是链接的?

（2）设有一个最小堆，即堆中任意结点的关键码均大于它的左子女和右子女的关键码。其具有最大值的元素可能在什么地方?

答：（1）堆得存储是顺序的

（2）最大元素一定是叶子节点，在最下两层上

7.排序有各种方法，如堆排序、直接插入排序、冒泡排序、快速排序等方法。设一数组中原有数据如下：15，13，20，18，12，60。下面是一组由不同排序方法进行一遍排序后的结果。请在下面各题的（ ）中写出排序方法。

（ 快排 ）排序的结果为：12，13，15，18，20，60

（ 冒泡 ）排序的结果为：13，15，18，12，20，60

（ 直接插入）排序的结果为：13，15，20，18，12，60

（ 堆排序）排序的结果为：12，13，20，18，15，60

数据结构期末考点:

基于单链表的插删(带头结点)

栈的应用--简答题(不写算法)

两栈若共享空间(底设两端，相向运动)

栈满:两栈顶指针值相减的绝对值为1（或两栈顶指针相邻）

入栈出栈可能出现的情况.

队列判空、判满，指针修改(无大题)

树的层次遍历(图的广度搜索)、输入输出缓冲区(无大题)

串的基本概念(空串、空格串)--无算法

串的相关概念--子串、串相等、模式匹配

考得较少的部分:数组、广义表、矩阵的压缩、稀疏矩阵压缩方案名称

广义表:表长(表的元素个数)、表深(括号最大层数)、非空表的表头表尾--掌握概念

重点--线性表、栈、队列

读算法--二叉树五条基本性质、二叉链表的存储结构、先序建立

二叉排序树--建立、读算法、根据(根、左、右)画出二叉排序树

画二叉排序树--先序+中序，中序+后序

平均查找长ASL--只需判断平衡与否即可

递归算法--结点个数、深度

线索二叉树的目的

Huffman(定义、性质)(简答题)--统计字符，从下往上构造Huffman树

画出Huffman树(习题册练习):编码、WPL(出现频率\*权)

邻接矩阵:给定图

邻接表(简答题):深搜、广搜--从某点开始进行

图的连通性--强、弱，顺序

算法的基本思想:画最小生成树--简答题:Prim/Kruscal算法 判断

拓扑排序:简答题，选择题--有向网写拓扑排序的基本思想

关键路径:活动AOE 基本概念

Dijistra算最短路径(无大算法)

查找:顺序查找--(n+1)/2 折半查找--log2(n+1)-1

哈希--除留余数法、开放地址法(线性勘测再散列、二次勘测再散列)、链地址法

排序:内部排序 时间复杂度(一般为最坏情况)、稳定性(由方法决定，位置是否发生变化)