1．下面程序段用于求两个n\*n矩阵相乘的算法，试求其时间复杂度。

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

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

c[i][j]=0;

for(k=0;k<n;k++)

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

}

2．求下述最小函数渐近时间。

（1）T1(n)=nlogn+n

（2）T2(n)=nlogn-n

（3）T3(n)=n3-100logn

3．设A是一个线性表（al，a2，…,an），采用顺序存储结构，则在等概率的前提下，平均每插入一个元素需要移动的元素个数为多少？

4．在单链表和双向链表中，能否从当前结点出发访问到任一结点？

5．已知一个栈S的输入序列为abcd，下面两个序列能否通过栈的Push和Pop操作输出；如果能，请写出操作序列；如果不能，说明原因。

（1）dbca （2）cbda

6．线性表可用顺序表或链表存储。试问：

(1) 两种存储表示各有哪些主要优缺点?

(2) 如果有n个表同时并存，并且在处理过程中各表的长度会动态发生变化，表的总数也可能自动改变、在此情况下，应选用哪种存储表示？为什么？

(3) 若表的总数基本稳定，且很少进行插入和删除，但要求以最快的速度存取表中的元素，这时，应采用哪种存储表示？为什么？

7. 试证明：若借助栈可由输入序列1, 2, 3, …, n得到一个输出序列p1, p2, p3, …, pn (它是输入序列的某一种排列)，则在输出序列中不可能出现以下情况，即存在i < j < k，使得pj < pk < pi。(提示：用反证法)

1.由程序可知包含三层循环且每层均为n，时间复杂度为O(n3)。

2. (1)O(nlogn)

(2)O(nlogn)

(3)O(n3)

3.(n+(n-1)+(n-2)+……+2+1+0)/(n+1) = n/2

平均移动 n/2 次

4.一般情况下单链表不行，双向链表可以，但若该单链表为一个循环链表则也可以。

5.（1）不能

若输出的第一个字母为d说明序列已经全按序压入栈中，第二个输出元素应为c

（2） 能

push() push() push() pop() pop() push() pop() pop()

6.

（1）线性表在访问某一位置的元素时时间复杂度为O（1），但在插入或删除的时的时间复杂度为O（n）。链表与其相反，在访问某一位置的元素时时间复杂度为O（n），但在但在插入或删除的时的时间复杂度为O（1）。

（2）链表 因为表长度与数目不定，并存时使用链表结构只需将其连接即可，时间复杂度远低于线性表。

（3）顺序表 因为总数基本稳定，且很少进行插入和删除并要以最快的速度存取表中的元素，此时正符合顺序表的特点。

7.假设存在i< j<k, 使得pj<pk<pi。

考虑栈的性质，由于i<j<k, 所以pi先进栈，然后pj，最后pk；由于是顺序输入，对pj<pk<pi，可知pi在pj和pk之后进栈，与前者pi先进栈相矛盾，所以假设错位，故不存在i < j < k，使得pj < pk < pi。