一选择

1.以下不可以使用分治法求解的是（ D ）。

A.棋盘覆盖问题 B.选择问题 C.归并排序 D.0/1背包问题

二填空

1.动态规划算法的基本要素是\_\_\_\_\_\_\_，\_\_\_\_\_\_\_\_。

答案:最优子结构，重叠子问题

1. .贪心算法的基本要素是 贪心选择性 质和 最优子结构 性质 。
2. 算法的复杂性有 时间 复杂性和 空间 复杂性之分.
3. 从分治法的一般设计模式可以看出，用它设计出的程序一般是 递归算法 。
4. 快速排序在最坏情况下的时间复杂度为 O（n²）最好情况下的时间复杂度O（nlogn）
5. 若序列X={B,C,A,D,B,C,D}，Y={A,C,B,A,B,D,C,D}，请给出序列X和Y的一个最长公共子序列 {BABCD}或{CABCD}或{CADCD｝

三大题

1.试写出用分治法对一个有序表实现二分搜索的算法。

解：Template<class>

int BinarySearch(Type a[],const Type &x,int n)

{

int left=0,right=n-1;

while(left<=right)

{

int middle=(left+right)/2;

if(x==a[middle]) return middle+1;

if(x>a[middle]) left=middle+1;

else right=middle-1;

}

return -1;

}

2.写出设计动态规划算法的主要步骤。

答案：

1，问题具有最优子结构性质

2，构造最优值的递归关系表达式

3，最优值的算法描述

4，构造最优解

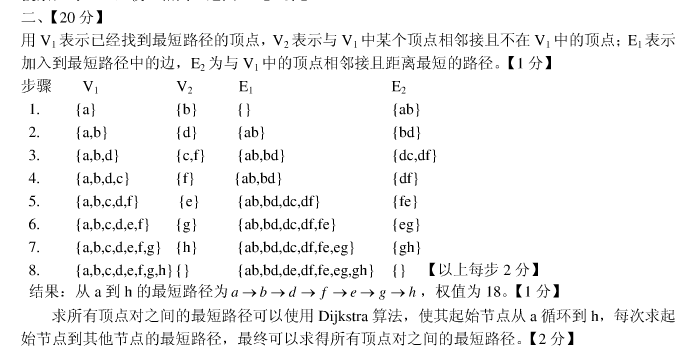
3.对下列各组函数f (n) 和g (n)，确定f (n) = O (g (n)) 或f (n) =Ω(g (n))或f(n) =

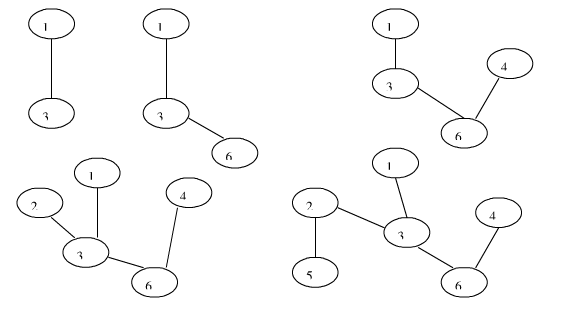
θ(g(n))，并简要说明理由。

(1) f(n)=2n； g(n)=n! (2) f(n)=n； g (n)=log n2 (3) f(n)=100； g(n)=log100 (4) f(n)=n3； g(n)= 3n (5) f(n)=3n； g(n)=2n 答：

(1) f(n) = O(g(n)) 因为g(n)的阶比f(n)的阶高。 (2) f(n) = Ω(g(n)) 因为g(n)的阶比f(n)的阶低。 (3) f(n) = θ(g(n)) 因为g(n)与f(n)同阶。 (4) f(n) = O(g(n)) 因为g(n)的阶比f(n)的阶高。 (5) f(n) = Ω(g(n)) 因为g(n)的阶比f(n)的阶低。

4.

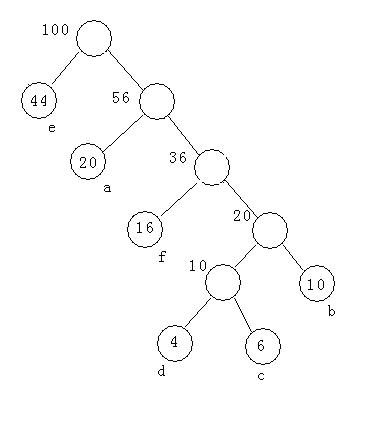




**5.考虑用哈夫曼算法来找字符a,b,c,d,e,f 的最优编码。这些字符出现在文件中**

**的频数之比为 20:10:6:4:44:16。构造对应的哈夫曼树，并据此给出a,b,c,d,e,f 的一种最优编码。**

解：根据题中数据构造哈夫曼树如下图所示。



由此可以得出 a,b,c,d,e,f 的一组最优的编码：01,0000,00010,00011, 1,001。

6.求解特征方程

f (n)=7f(n-1)-16f(n-2)+12f(f-3)

f (0)=2

f (1)=4

f (2)=10

解:

特征方程为x³-7x²+16x-12=0

可以改写为X³-3X²-4X²+12X+4X-12=0

（X-3）（X²-4X-4）=0

（X-2）（X-2）(X-3)=0

所以q1=2,q2=2,q3=3

所以递推方程的通解为

f（n）=(c1+c2 n)q1 n +c3 q3 n

= c1 \*2 n+ c2n 2 n +c3 3 n

代入初始条件：

f(0)=c1+c3=2

f(1)=2c1+2c2+3c3=4

f(2)=4c1+8c2+9c3=10

解得 c1=0 ,c2= -1 c3=2

则递归方程的解为 f(n)= (c1+c2 n)q1 n +c3 q3 n =3 n+1 – n\*2 n

7.编写一个算法，可以检测一个字符串是否回文（如：afaddafa，abwba等）。 int fun(char \*A,int n){ int i ,j; i=0,j=n-1; while(i<=j){

if(A[i]!=A[j])break;

i++;j--; }

if(i<=j)return 0; else return 1; }