分治法

1、二分搜索算法是利用（　分治策略）实现的算法。

9. 实现循环赛日程表利用的算法是（分治策略 ）

27、Strassen 矩阵乘法是利用（分治策略　）实现的算法。

34. 实现合并排序利用的算法是（分治策略 ）。实

现大整数的乘法是利用的算法（　　分治策略 ）。

17. 实现棋盘覆盖算法利用的算法是（分治法 ）。

29、使用分治法求解不需要满足的条件是（子问题必须是一样的 ）。

不可以使用分治法求解的是（0/1 背包问题 ）。

动态规划

下列不是动态规划算法基本步骤的是（ 构造最优解 ）

下列是动态规划算法基本要素的是（子问题重叠性质 ）。

下列算法中通常以自底向上的方式求解最优解的是（动态规划法　）

备忘录方法是那种算法的变形。（　动态规划法　　）

最长公共子序列算法利用的算法是（　动态规划法 ）。矩

阵连乘问题的算法可由（动态规划算法 B）设计实现。实

现最大子段和利用的算法是（　　动态规划法　　）。

贪心算法

能解决的问题：单源最短路径问题，最小花费生成树问题，背包问题，活动安排问题，

不能解决的问题：N 皇后问题，0/1 背包问题

是贪心算法的基本要素的是（贪心选择性质和最优子结构性质）。

回溯法

回溯法解旅行售货员问题时的解空间树是（　排列树 ）。

剪枝函数是回溯法中为避免无效搜索采取的策略

回溯法的效率不依赖于下列哪些因素（　确定解空间的时间）

分支限界法

最大效益优先是（　分支界限法 ）的一搜索方式。

分支限界法解最大团问题时，活结点表的组织形式是（　最大堆 ）。分支限界法解旅行售货员问题时，活结点表的组织形式是（ 最小堆 ）

优先队列式分支限界法选取扩展结点的原则是（ 结点的优先级 ）

在对问题的解空间树进行搜索的方法中,一个活结点最多有一次机会成为活结点的是(　分支限界法).

从活结点表中选择下一个扩展结点的不同方式将导致不同的分支限界法,以下除(　栈式分支限界法)之外都是最常见的方式.

（1）队列式(FIFO)分支限界法：按照队列先进先出（FIFO）原则选取下一个节点为扩展节点。

（2）优先队列式分支限界法：按照优先队列中规定的优先级选取优先级最高的节点成为当前扩展节点。

（最优子结构性质）是贪心算法与动态规划算法的共同点。

贪心算法与动态规划算法的主要区别是（　贪心选择性质　　）。

回溯算法和分支限界法的问题的解空间树不会是(　无序树　).

14．哈弗曼编码的贪心算法所需的计算时间为（　　B　　）。

A、O（$n2^n$）　　B、O（nlogn）　　C、O（2^n）　　D、O（n）

21、下面关于 NP 问题说法正确的是（B ）

A NP 问题都是不可能解决的问题

B P 类问题包含在 NP 类问题中

C NP 完全问题是 P 类问题的子集

D NP 类问题包含在 P 类问题中

40、背包问题的贪心算法所需的计算时间为（　　B　　）

A、O（$n2^n$）　　B、O（nlogn）　　C、O（2^n）　　D、O（n）

42. 0-1 背包问题的回溯算法所需的计算时间为（　A　）

A、O（n2n）　B、O（nlogn）　C、O（2n）　D、O（n）

.

47.背包问题的贪心算法所需的计算时间为（　B　）。

A、O（$n2^n$）　B、O（nlogn）　C、O（2^n）　D、O（n）

53. 采用贪心算法的最优装载问题的主要计算量在于将集装箱依其重量从小到大排序，故算法的时间复杂度为　（　B　）。

A、O（$n2^n$）　B、O（nlogn）　C、O（2^n）　D、O（n）

56、算法是由若干条指令组成的有穷序列，而且满足以下性质（ D　）

（1）　输入：有 0 个或多个输入

（2）　输出：至少有一个输出

（3）　确定性：指令清晰，无歧义

（4）　有限性：指令执行次数有限，而且执行时间有限

A (1)(2)(3)　　B(1)(2)(4)　　C(1)(3)(4)　　D (1) (2)(3)(4)

57、函数 $32^n+10n\log^n$ 的渐进表达式是(B　).

A. 2^n　　B. 32^n　　C. $n\log^n$　　D. $10n\log^n$

59、用动态规划算法解决最大字段和问题，其时间复杂性为(B　).

A.logn　B.n　C.n^2　D.nlogn

61、设 f(N),g(N)是定义在正数集上的正函数,如果存在正的常数 C 和自然数 N_0,使得当 $N\geq N_0$ 时有 $f(N)\leq Cg(N)$,则称函数 f(N)当 N 充分大时有下界 g(N),记作 $f(N)\in\Omega(g(N))$,即 f(N)的阶(　A　)g(N)的阶.

A.不高于　B.不低于　C.等价于　D.逼近

二、　填空题

2、程序是<u>　算法　</u>用某种程序设计语言的具体实现。

3、算法的“确定性”指的是组成算法的每条<u>　指令　</u>是清晰的，无歧义的。

6、算法是指解决问题的<u>　一种方法　</u>或<u>　一个过程　</u>。

7、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　。

11、计算一个算法时间复杂度通常可以计算 循环次数 、 基本操作的频率或计算步。

14、解决 0/1 背包问题可以使用动态规划、回溯法和分支限界法，其中不需要排序的是 动态规划 ，需要排序的是 回溯法 ，分支限界法 。

15、使用回溯法进行状态空间树裁剪分支时一般有两个标准：约束条件和目标函数的界，N 皇后问题和 0/1 背包问题正好是两种不同的类型，其中同时使用约束条件和目标函数的界进行裁剪的是 0/1 背包问题 ，只使用约束条件进行裁剪的是 N 皇后问题 。

30.回溯法是一种既带有 系统性 又带有 跳跃性 的搜索算法。

33. 回溯法搜索解空间树时，常用的两种剪枝函数为 约束函数 和 限界函数 。

34. 任何可用计算机求解的问题所需的时间都与其 规模 有关。

35. 快速排序算法的性能取决于 划分的对称性 。

36. Prim 算法利用 贪心 策略求解 最小生成树 问题，其时间复杂度是 $O(n^2)$ 。

37. 图的 m 着色问题可用 回溯 法求解，其解空间树中叶子结点个数是 m^n ，解空间树中每个内结点的孩子数是 m 。

4.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}。

5.用回溯法解问题时，应明确定义问题的解空间，问题的解空间至少应包含一个（最优）解

8.0-1 背包问题的回溯算法所需的计算时间为 $o(n*2^n)$,用动态规划算法所需的计算时间为 $o(\min\{nc,2^n\}$ 。

二、综合题（50 分）

1.写出设计动态规划算法的主要步骤。

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；

2.流水作业调度问题的 johnson 算法的思想。

①令 $N_1=\{i|a_i<b_i\}$，$N_2=\{i|a_i>=b_i\}$；②将 N_1 中作业按 a_i 的非减序排序得到 N_1’，将 N_2 中作业按 b_i 的非增序排序得到 N_2’；③N_1’中作业接 N_2’中作业就构成了满足 Johnson 法则的最优调度。

3. 若 n=4，在机器 M1 和 M2 上加工作业 i 所需的时间分别为 a_i 和 b_i，且 $(a_1,a_2,a_3,a_4)=(4,5,12,10)$，$(b_1,b_2,b_3,b_4)=(8,2,15,9)$求 4 个作业的最优调度方案，并计算最优值。

步骤为：N1={1，3}，N2={2，4}；

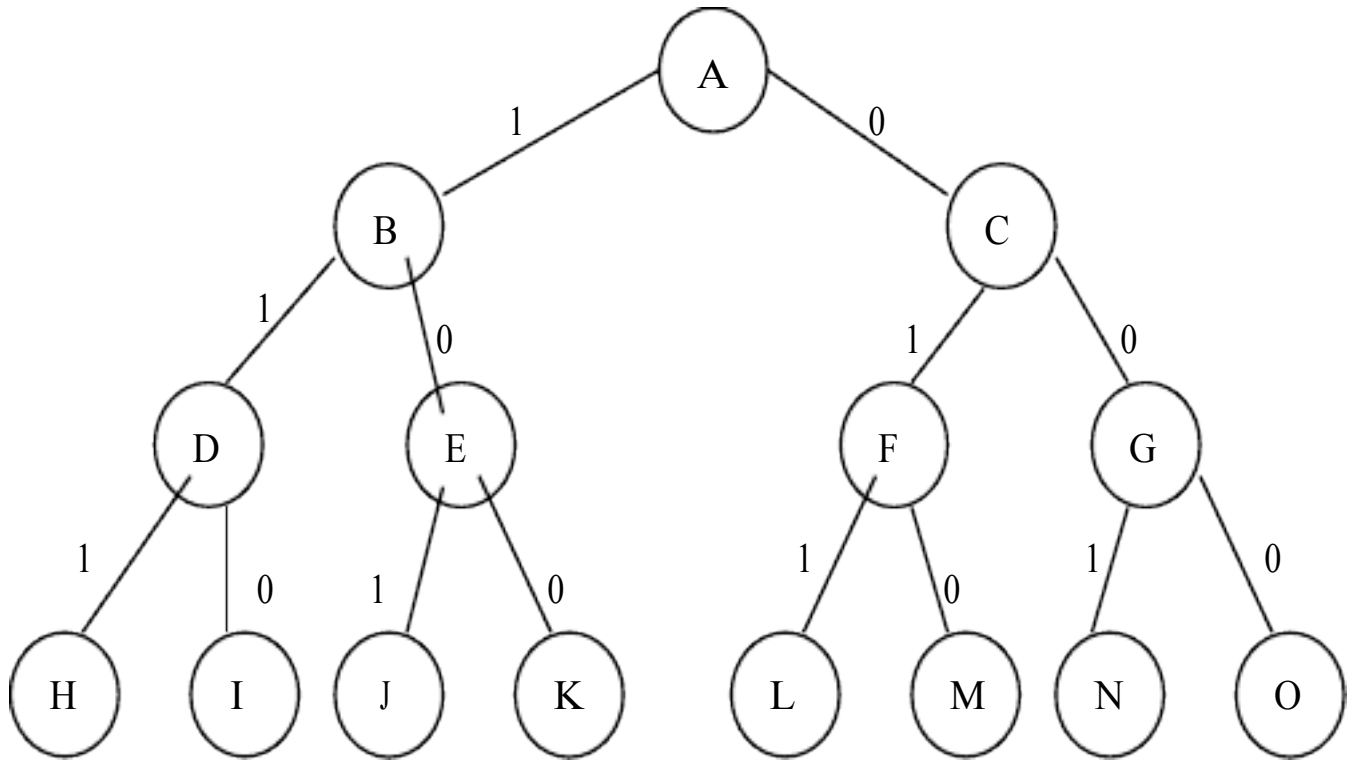
N_1'={1，3}，N_2'={4，2}；

最优值为：38

4.使用回溯法解 0/1 背包问题：n=3，C=9，V={6,10,3}，W={3,4,4},其解空间有长度为 3 的0-1 向量组成，要求用一棵完全二叉树表示其解空间（从根出发，左 1 右 0），并画出其解空间树，计算其最优值及最优解。

解空间为{(0,0,0),(0,1,0),(0,0,1),(1,0,0),(0,1,1),(1,0,1),
(1,1,0),(1,1,1)}。

解空间树为：



该问题的最优值为：16　　最优解为:（1，1，0）

5.设 S=｛X_1，X_2，•••，X_n｝是严格递增的有序集，利用二叉树的结点来存储 S 中的元素，在表示 S 的二叉搜索树中搜索一个元素 X，返回的结果有两种情形,（1）在二叉搜索树的内结点中找到 $X=X_i$，其概率为 b_i。（2）在二叉搜索树的叶结点中确定 $X\in(X_i, X_{i+1})$，其概率为 a_i。在表示 S 的二叉搜索树 T 中，设存储元素 X_i 的结点深度为 C_i；叶结点（X_i，X_{i+1}）的结点深度为d_i，则二叉搜索树T 的平均路长 p 为多少？假设二叉搜索树 $T[i][j]$= ｛X_i，X_{i+1}，•••，X_j｝最优值为 m[i][j]，$W[i][j]=a_{i-1}+b_i+\bullet\bullet\bullet+b_j+a_j$，则 m[i][j](1<=i<=j<=n)递归关系表达式为什么？

二叉树 T 的平均路长 $P=\sum_{i=1}^{n} bi*(1+Ci)+\sum_{j=0}^{n} aj*dj$

{

m[i][j]=W[i][j]+min{m[i][k]+m[k+1][j]} (1<=i<=j<=n,m[i][i-1]=0)

m[i][j]=0　(i>j)

6.描述 0-1 背包问题。

已知一个背包的容量为 C，有 n 件物品，物品i 的重量为 W_i，价值为V_i，求应如

何选择装入背包中的物品，使得装入背包中物品的总价值最大。

三、简答题（30 分）

1.流水作业调度中，已知有 n 个作业，机器M1 和 M2 上加工作业 i 所需的时间分别为 a_i 和 b_i，请写出流水作业调度问题的 johnson 法则中对 a_i 和 b_i 的排序算法。（函数名可写为 sort(s,n)）

2.最优二叉搜索树问题的动态规划算法（设函数名 binarysearchtree)） 1.

```
void sort(flowjope s[],intn)
 {
  int i,k,j,l;
  for(i=1;i<=n-1;i++)//----------选择排序
   {
    k=i;
    while(k<=n&&s[k].tag!=0)        k++;
    if(k>n)    break;//-----没有 a_i，跳出else
    {for(j=k+1;j<=n;j++)
        if(s[j].tag==0)
        if(s[k].a>s[j].a)       k=j;
     swap(s[i].index,s[k].index);
     swap(s[i].tag,s[k].tag); }
  }
 l=i;//-----记下当前第一个 b_i 的下标
 for(i=l;i<=n-1;i++)
  {
   k=i;
   for(j=k+1;j<=n;j++)
      if(s[k].b<s[j].b)     k=j;
   swap(s[i].index,s[k].index); //-----只移动  index 和  tag
   swap(s[i].tag,s[k].tag);                    }
}
```

2.

```
void binarysearchtree(int a[],int b[],int n,int **m,int **s,int **w)
 {
int i,j,k,t,l;
for(i=1;i<=n+1;i++)
{ w[i][i-1]=a[i-1];
   m[i][i-1]=0;}
for(l=0;l<=n-1;l++)//----l 是下标  j-i 的差
  for(i=1;i<=n-l;i++)
    {   j=i+l;
        w[i][j]=w[i][j-1]+a[j]+b[j];
        m[i][j]=m[i][i-1]+m[i+1][j]+w[i][j];
        s[i][j]=i;
```

```
            for(k=i+1;k<=j;k++)
              {     t=m[i][k-1]+m[k+1][j]+w[i][j];
                    if(t<m[i][j])
                    { m[i][j]=t;
                       s[i][j]=k;
                     }
            }
      }
}
```

一、　填空题（本题 15 分，每小题 1 分）

1、算法就是一组有穷的 规则 ，它们规定了解决某一特定类型问题的 一系列运算

2、在进行问题的计算复杂性分析之前，首先必须建立求解问题所用的计算模型。3 个基本计算模型是 随机存取机 RAM 、 随机存取存储程序机RASP 、 图 灵 机 。

3、算法的复杂性是 算法效率 的度量，是评价算法优劣的重要依据。

4、计算机的资源最重要的是 时间 和 空间 资源

5、$f(n)=6\times2^n+n^2$，f(n)的渐进性态 f(n)= O(2^n)

6、贪心算法总是做出在当前看来 最好 的选择。也就是说贪心算法并不从整体最优考虑，它所做出的选择只是在某种意义上的局部最优结构

二、简答题（本题 25 分，每小题 5 分）

1、　简单描述分治法的基本思想。

2、　简述动态规划方法所运用的最优化原理。

3、　何谓最优子结构性质？

4、　简单描述回溯法基本思想。

5、　何谓 P、NP、NPC 问题

三、算法填空（本题 20 分，每小题 5 分）

1、n 后问题回溯算法

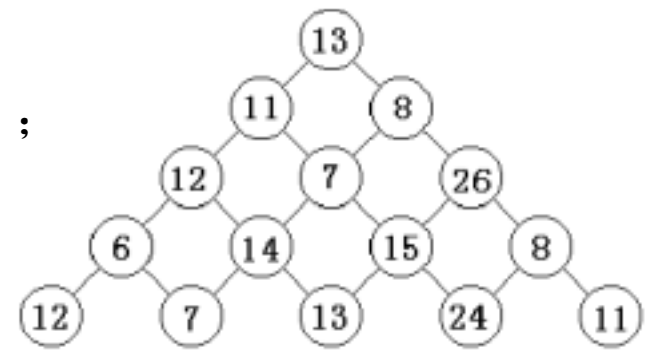
(1)用二维数组A[N][N]存储皇后位置,若第i 行第j 列放有皇后,则A[i][j]为非 0 值,否则值为 0。

(2)分别用一维数组 M[N]、L[2*N-1]、R[2*N-1]表示竖列、左斜线、右斜线是否放有棋子，有则值为 1,否则值为 0。

```
for(j=0;j<N;j++)
 if(  1     )  /*安全检查*/
      { A[i][j]=i+1;     /*放皇后*/
        2     ;
        if(i==N-1)    输出结果;
        else    3     ;  ;       /*试探
        下一行*/ 4     ;        /*去皇
            后*/
            5     ;  ;
```

}

2、数塔问题。有形如下图所示的数塔，从顶部出发，在每一结点可以选择向左走或是向右走，一起走到底层，要求找出一条路径，使路径上的值最大。




```
for(r=n-2;r>=0;r--) //自底向上递归计算
 for(c=0;     1      ;c++)
  if( t[r+1][c]>t[r+1][c+1])     2     ;
    else     3          ;
```

3、Hanoi 算法

```
 Hanoi(n,a,b,c)
 if （n==1）     1      ;
 else
 { 	2      ;
      3      ;
   Hanoi(n-1,b, a, c);
 }
```

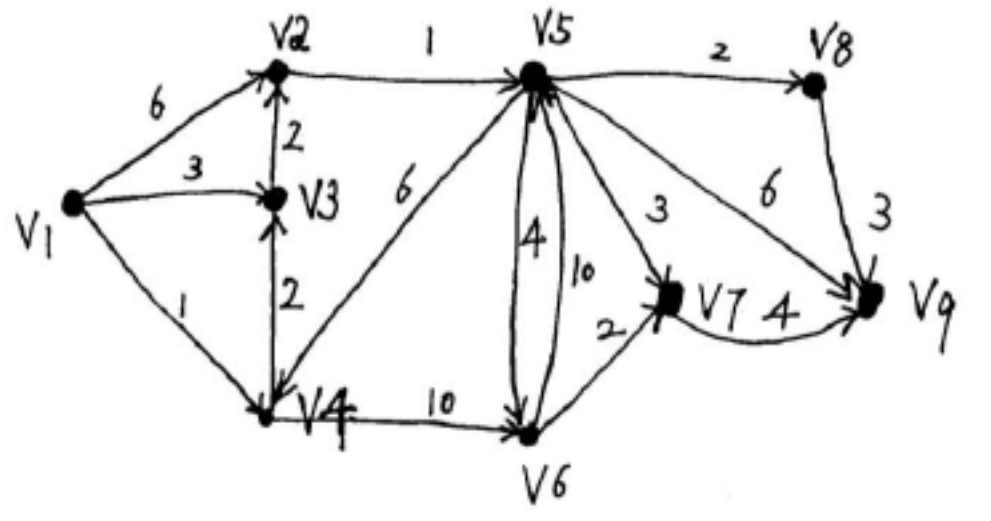
4、Dijkstra 算法求单源最短路径

d[u]:s 到 u 的距离　p[u]:记录前一节点信息

```
Init-single-source(G,s)
for each vertex v∈V[G]
 do { d[v]=∞;   1       }
 d[s]=0
Relax(u,v,w)
if d[v]>d[u]+w(u,v)
then { d[v]=d[u]+w[u,v];
        2
       }
dijkstra(G,w,s)
1. Init-single-source(G,s)
2. S=Φ
3. Q=V[G]
4.while Q<> Φ
  do u=min(Q)
     S=S∪{u}
     for each vertex     3
       do     4
```

四、算法理解题（本题 10 分）

根据优先队列式分支限界法，求下图中从v1 点到v9 点的单源最短路径，请画出求得最优解的解空间树。要求中间被舍弃的结点用×标记，获得中间解的结点用单圆圈○框起，最优解用双圆圈◎框起。



五、算法理解题（本题 5 分）

设有 $n=2^k$ 个运动员要进行循环赛，现设计一个满足以下要求的比赛日程表：

①每个选手必须与其他 n-1 名选手比赛各一次；

②每个选手一天至多只能赛一次；

③循环赛要在最短时间内完成。

（1）如果 $n=2^k$，循环赛最少需要进行几天；

（2）当 $n=2^3=8$ 时，请画出循环赛日程表。

六、算法设计题（本题 15 分）

分别用贪心算法、动态规划法、回溯法设计 0-1 背包问题。要求：说明所使用的算法策略；写出算法实现的主要步骤；分析算法的时间。

七、算法设计题（本题 10 分）

通过键盘输入一个高精度的正整数 n(n 的有效位数≤240)，去掉其中任意 s 个数字后，剩下的数字按原左右次序将组成一个新的正整数。编程对给定的 n 和s，寻找一种方案，使得剩下的数字组成的新数最小。

【样例输入】

178543

S=4

【样例输出】

13

二、简答题（本题 25 分，每小题 5 分）

6、**分治法的基本思想**是将一个规模为n 的问题分解为k 个规模较小的子问题，这些子问题互相独立且与原问题相同；对这 k 个子问题分别求解。如果子问题的规模仍然不够小，则再划分为 k 个子问题，如此递归的进行下去，直到问题规模足够小，很容易求出其解为止；将求出的小规模的问题的解合并为一个更大规模的问题的解，自底向上逐步求出原来问题的解。

7、“**最优化原理**”用数学化的语言来描述：假设为了解决某一优化问题，需要依次作出 n 个决策 D1，D2，...，Dn，如若这个决策序列是最优的，对于任何一个整数 k，1 ＜ k ＜ n，不论前面 k 个决策是怎样的，以后的最优决策只取决于由前面决策所确定的当前状态，即以后的决策 Dk+1，Dk+2，...，Dn 也是最优的。

8、某个问题的最优解包含着其子问题的最优解。这种性质称为 **最优子结构性质**。

9、**回溯法的基本思想**是在一棵含有问题全部可能解的状态空间树上进行深度优先搜索，解为叶子结点。搜索过程中，每到达一个结点时，则判断该结点为根的子树是否含有问题的解，如果可以确定该子树中不含有问题的解，则放弃对该子树的搜索，退回到上层父结点，继续下一步深度优先搜索过程。在回溯法中，并不是先构造出整棵状态空间树，再进行搜索，而是在搜索过程，逐步构造出状态空间树，即边搜索，边构造。

10、 P(Polynomial 问题)：也即是多项式复杂程度的问题。

NP 就是 Non-deterministic Polynomial 的问题，也即是多项式复杂程度的非确定性问题。

NPC(NP Complete)问题，这种问题只有把解域里面的所有可能都穷举了之后才能得出答案，这样的问题是NP 里面最难的问题，这种问题就是NPC 问题。

三、算法填空（本题 20 分，每小题 5 分）

1、n 后问题回溯算法

(1) !M[j]&&!L[i+j]&&!R[i-j+N]

(2) M[j]=L[i+j]=R[i-j+N]=1;

(3) try(i+1,M,L,R,A)

(4) A[i][j]=0

(5) M[j]=L[i+j]=R[i-j+N]=0

2、数塔问题。

（1）c<=r

(2)t[r][c]+=t[r+1][c]

（3）t[r][c]+=t[r+1][c+1]

3、Hanoi 算法

（1）move(a,c)

（2）Hanoi(n-1, a, c , b)

（3）Move(a,c)

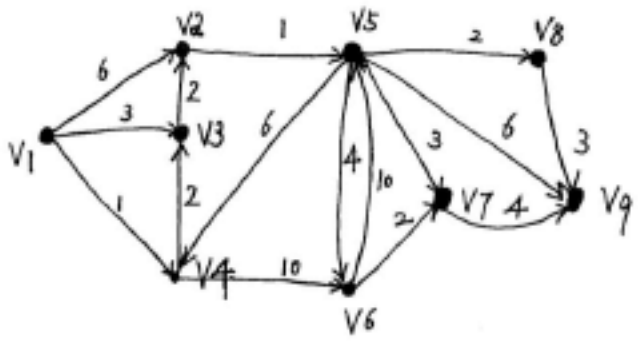
4、（1）p[v]=NIL

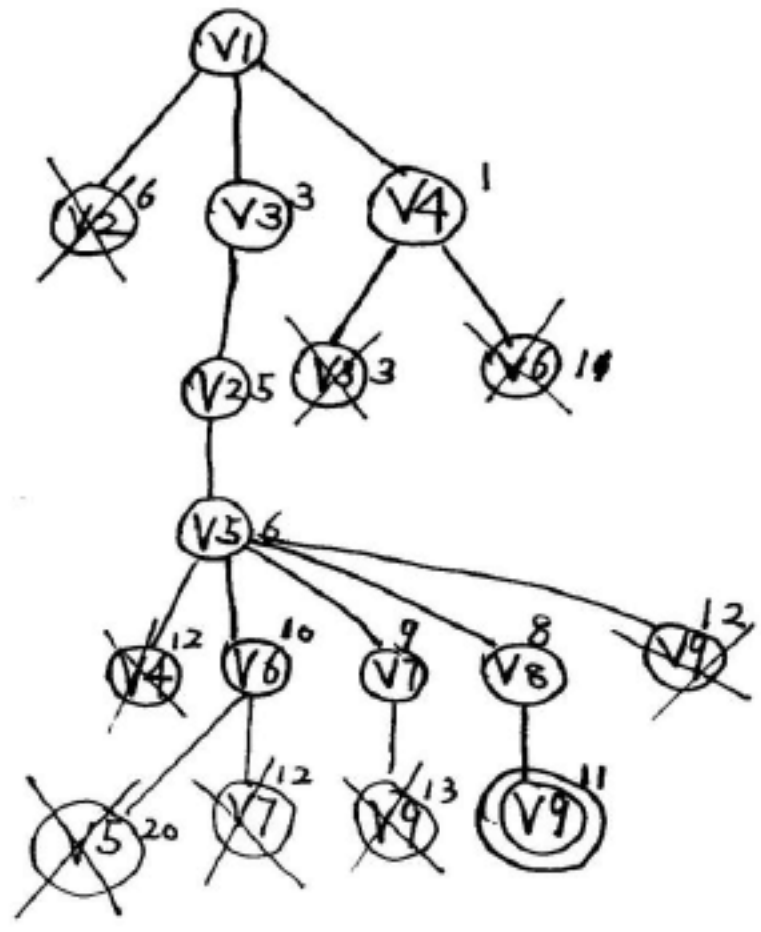
（2）p[v]=u

(3) v∈adj[u]

（4）Relax(u,v,w)

四、算法理解题（本题 10 分）





五、（1）8 天（2 分）；

（2） 当 n=23=8 时，循环赛日程表（3 分）。

1 2 3 4 2 1 4 3 3 4 1 2 4 3 2 1	5 6 7 8 6 5 8 7 7 8 5 6 8 7 6 5
5 6 7 8 6 5 8 7 7 8 5 6 8 7 6 5	1 2 3 4 2 1 4 3 3 4 1 2 4 3 2 1

六、算法设计题（本题 15 分）

（1）贪心算法 O（nlog（n））