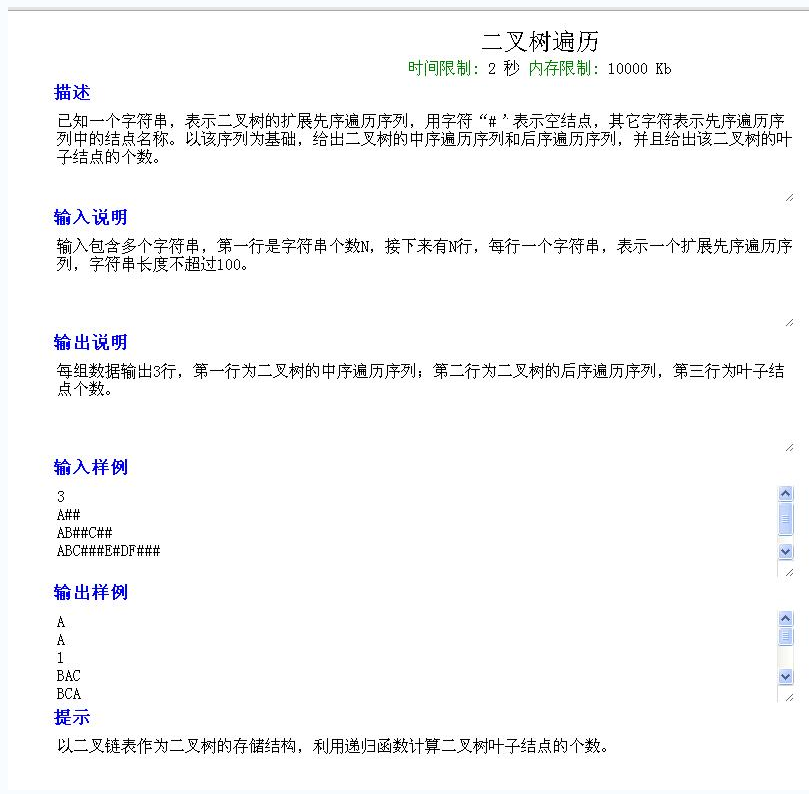
Exam1.cpp

最短路径  
描述：  
求图中任意两个顶点之间的最短路径。  
输入：  
输入数据第一行是一个正整数，表示图中的顶点个数n（顶点将分别按0，1，…，n-1进行编号）。之后的n行每行都包含n个整数，第i行第j个数表示顶点i-1和顶点j-1之间的边长，用10000来表示两个顶点之间无边。后面每行2个数字，表示一对待求最短路径的顶点，用-1 -1表示输入结束，-1 -1不求解。  
输出：  
每对待求最短路径的顶点输出两行数据：第一行输出两个顶点间的最短路径长度，第二行输出最短路径，要按顺序输出顶点编号序列，顶点间用空格隔开。当两个顶点间没有路径时，只在一行上输出字符串“NO”。  
  
示例输入  
7  
0 12 10000 10000 10000 10000 10000  
12 0 10000 10000 3 10000 10000  
10000 10000 0 10000 10000 21 11  
10000 10000 10000 0 10000 10000 10000  
10000 3 10000 10000 0 10000 8  
10000 10000 21 10000 10000 0 10000  
10000 10000 11 10000 8 10000 0  
0 2  
0 3  
5 0  
2 1  
1 5  
-1 -1  
示例输出  
34  
0 1 4 6 2  
NO  
55  
5 2 6 4 1 0  
22  
2 6 4 1  
43  
1 4 6 2 5  
提示：可以用Floyd算法实现，也可以在Dijkstra算法外面增加一层循环来实现。

Exam2.cpp

无向图的深度优先搜索  
问题描述  
已知无向图的邻接矩阵，以该矩阵为基础，给出深度优先搜索遍历序列，并且给出该无向图的连通分量的个数。  
在遍历时，当有多个顶点可选时，优先选择编号小的顶点。  
输入说明：  
第一行是1个正整数，为顶点个数n，顶点编号依次为0，1，…，n-1。后面是邻接矩阵，n行n列。  
输出说明：  
共2行。第一行输出为无向图的深度优先搜索遍历序列，输出为顶点编号，顶点编号之间用空格隔开；第二行为无向图的连通分量的个数。  
输入样例：  
6  
0 1 0 0 0 0  
1 0 0 0 1 0  
0 0 0 1 0 0  
0 0 1 0 0 0  
0 1 0 0 0 1  
0 0 0 0 1 0  
输出样例  
0 1 4 5 2 3  
2  
提示  
以邻接矩阵为存储结构，进行深度优先搜索遍历，在遍历的过程中计算无向图连通分量的个数。

19年考题：输出连通分量个数

  
3

A##

AB##C##

ABC###E#DF###  
 exam3.cpp  
  


Exam4.c

19年考题：

1.给两个数组，输出数组交集以外的数字

2.连通分量个数

3.给出前缀表达式 输出结果

4.还原二叉树 给出前中序遍历，输出后序遍历 exam5.cpp