**算法训练 黑白无常**

时间限制：1.0s   内存限制：256.0MB

问题描述

　　某寝室的同学们在学术完之后准备玩一个游戏：游戏是这样的，每个人头上都被贴了一张白色或者黑色的纸，现在每个人都会说一句话“我看到x张白色纸条和y张黑色的纸条”，又已知每个头上贴着白色纸的人说的是真话、每个头上贴着黑色纸的人说的是谎话，现在要求你判断哪些人头上贴着的是白色的纸条，如果无解输出“NoSolution.”；如果有多组解，则把每个答案中贴白条的人的编号按照大小排列后组成一个数（比如第一个人和第三个人头上贴着的是白纸条，那么这个数就是13；如果第6、7、8个人都贴的是白纸条，那么这个数就是678）输出最小的那个数（如果全部都是黑纸条也满足情况的话，那么输出0）

输入格式

　　第一行为一个整数n，接下来n行中的第i行有两个整数x和y，分别表示第i个人说“我看到x张白色纸条和y张黑色的纸条”。

输出格式

　　一行。如果无解输出“NoSolution.”。否则输出答案中数值（具体见问题描述）最小的那个，如果全部都是黑纸条也满足情况的话，那么输出0

样例输入

2  
1 0  
1 0

样例输出

0

样例输入

5  
3 1  
0 4  
1 3  
4 0  
1 3

样例输出

35

数据规模和约定

n<=8

本题的C++参考代码如下：

#include<stdio.h>

#include<string.h>

#include<math.h>

int mt[9][2],n;

int max[9]={9};

int xl[9]={9};

bool ifc();

int main()

{

int i,j,m,fi,k,flag;

scanf("%d",&n);

for(i=1;i<=n;i++) scanf("%d %d",&mt[i][0],&mt[i][1]);

m=pow(2,n);

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

{

for(fi=i,j=1,xl[0]=0;j<=n;j++)

{

xl[j]=fi&1;

fi>>=1;

if(xl[j]) xl[0]++;//真话数量

}

if(xl[0]>max[0]) continue;

if(ifc())//成立

{

// for(k=1;k<=n;k++) printf("%d",xl[k]);

// printf(" :%d\n",xl[0]);

if(max[0]==xl[0])

{

for(flag=k=1;k<=n;k++)

{

if(max[k]>xl[k])

{

flag=0;

break;

}

else if(max[k]<xl[k])

{

break;

}

}

if(!flag) continue;

}

for(k=0;k<=n;k++) max[k]=xl[k];

}

}

if(max[0]==9) printf("NoSolution.\n");

else if(max[0]==0) printf("0\n");

else

{

for(i=1;i<=n;i++) if(max[i]) printf("%d",i);

putchar('\n');

}

}

bool ifc()

{

int i;

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

{

if(xl[i])//若此人说真话

{

if(mt[i][0]!=xl[0]-1) return false;

}

else

{

if(mt[i][0]==xl[0]) return false;

}

}

return true;

}

本题的C参考代码如下：

#include <stdio.h>

#include <math.h>

int n;

int x[8],y[8];

int isok(int a[],int num,int value) //第i个人的话是否成立 num为人下标 value 为 0白 或者 1黑

{

int black=0,white=0,i;

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

{

if(i!=num)

{

if(a[i]) black++; //黑色

else white++; //白色

}

}

if(value) //黑

{

if(x[num]==white && y[num]==black) return 0;

else return 1;

}

else //白

{

if(x[num]==white && y[num]==black) return 1;

else return 0;

}

}

int main()

{

int i,j,k,sign,min=90000000,num;

int a[8],b[8];

scanf("%d",&n);

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

scanf("%d%d",x+i,y+i);

for(i=0;i<pow(2,n);i++)

{

k=i;

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

{

a[j]=k%2;

k/=2;

}

sign=1;

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

{

if(isok(a,j,a[j])==0)

{

sign=0;

break;

}

}

if(sign)

{

k=1;

num=0;

for(j=n-1;j>=0;j--)

{

if(a[j]==0)

{

num+=k\*(j+1);

k\*=10;

}

}

if(num<min) min=num;

}

}

if(min==90000000) printf("NoSolution.");

else

{

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

b[i]=1;

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

{

if(isok(b,i,b[i])==0)

{

sign=0;

break;

}

}

if(sign==0) printf("%d",min);

else printf("0");

}

return 0;

}

本题的Java参考代码如下：

import java.util.Scanner;

public class Main{

public static void main(String args[]) {

Scanner scanner = new Scanner(System.in);

int[][] xy;

int n, x, y;

n = scanner.nextInt();

xy = new int[n + 1][2];

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

xy[i][0] = scanner.nextInt();

xy[i][1] = scanner.nextInt();

}

String[] result = new String[n + 1];

int flag = 1;

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

// 是否为全黑标志位

if (xy[i][0] == 0 && xy[i][1] == n - 1)

flag = 0;

result[i] = "" + i;

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

if(a == i)continue ;

if (xy[i][0] == xy[a][0] && xy[i][1] == xy[a][1]) {

if(xy[i][0] == 0){

result[0] = a<i ? ""+a : ""+i ;

}

if (result[i].length() < (xy[i][0]+1))

result[i] = result[i] + a;

}

}

}

if (flag == 1) {

System.out.println(0);

} else {

int temp = 100000000;

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

if (result[i].length() > 1) {

if(temp != 100000000 && String.valueOf(temp).length() < result[i].length()){

temp = Integer.parseInt(result[i]) ;

}else{

temp = temp > Integer.parseInt(result[i]) ? Integer

.parseInt(result[i]) : temp;

}

}

}

if (temp != 100000000) {

System.out.println(temp);

} else if(result[0] != null){

System.out.println(result[0]) ;

}else{

System.out.println("NoSolution.") ;

}

}

}

}