**★★★☆☆**

**題組：Contest Volumes** **with Online Judge**

**題號：10099: The Tourist Guide**

**解題者：黃子芯**

**解題日期：2022年4月14日**

**題意：**

給定兩個正整數 N(<=100)、R，輸入0 0即為結束，N表示有N個景點，R則表示有R條路線，之後給定每條路線的起點、終點、限制人數，最後輸入要求的起始點及終點和導遊需帶領的人數，求出能夠以多少趟結束。

**題意範例：**

7 10

1 2 30

 1 3 15

 1 4 10

 2 4 25

 2 5 60

 3 4 40

 3 6 20

 4 7 35

 5 7 20

 6 7 30

 1 7 99

 0 0

→ Scenario #1

Minimum Number of Trips = 5

**解法：**

1. DFS-recursive(TLE)
2. DP

**解法範例：**

dp[][]為儲存人數的二維陣列。

利用dp[i][j]=max(dp[i][j],min(dp[i,k],dp[k,j]))找出最小路徑的最大值。

**討論：**

1. 每條路線的限制人數都還要扣掉導遊，因為每趟都需要導遊帶領。

**程式：**

#include <iostream>

using namespace std;

int main(){

 int N,R;

 int cases=1;

 int value;

 int c1,c2,p;

 int s,d;//start,destination

 int output[100]={0};

 cin>>N>>R;

 while(R!=0 || N!=0){

 int dp[N+1][N+1]={0};

 for(int i=0;i<R;i++){

 cin>>c1>>c2>>p;

 dp[c1][c2]=p-1;

 dp[c2][c1]=p-1;

 }

 for(int k=1;k<=N;k++){

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

 for(int j=1;j<=N;j++){

 dp[i][j]=max(dp[i][j],min(dp[i][k],dp[k][j]));

//找出從i到j能載的最大的最小值路線

 }

 }

 }

 cin>>s>>d>>p;

 value=dp[s][d];

 if(p%value!=0)//判斷若有餘額則加一

 value=p/value+1;

 else value=p/value;

 output[cases]=value;

 cases++;

 cin>>N>>R;

 }

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

 cout<<"Scenario #"<<i<<endl;

 cout<<"Minimum Number of Trips = "<<output[i]<<endl;

 cout<<endl;

 }

}