**求最大网络流总结**

**一个有向图，每条边都有最大可能的数据传输量c，要求一个点到某个点一次能传输的最大数据量，即为求最大网络流，每条边实际的数据传输量称为流量。**

**基本思路：使用贪心算法，每次找出能使总流量更大的边，将数据向那条边流。**

**Ford-fulkerson算法**

1. **读入所有有向边，并且加入流量为0的反向边**
2. **DFS查找所有点u的未被遍历的边(u,v)**
3. **如果满足这条边流量大于0，更新这条边以及它的反向边的流量（同时会DFS遍历v点）**
4. **重复2操作，直到所有点被遍历完成**

**算法复杂度：O(m+n)≈O(n^2)**

**Dinic算法**

**原理：利用BFS求最短路的方法将所有的边按从源点到汇点的距离排序，从而避免不必要的增广**

1. **BFS求源点到所有点的距离**
2. **DFS 遍历所有边(u,v)**
3. **如果dis[v]==dis[u]+1且边的长度大于0，那么更新这条边以及它的反向边的流量（同时DFS遍历了v点）**

**用dis数组的好处：不仅减少了不必要的增广步骤，而且使增广顺序有序，不必记录点是否已被遍历**

1. **重复2操作，直到没有符合条件的边为止**

**算法复杂度：**O(n^2\*m)。**（最坏情况，平均情况大大优于FF算法）**

**割**

**在网络中，将源点与汇点通过割边的方式划分成两个部分，割掉的所有边都是割，其中，s到t为正向割边，t到s为逆向割边**

**割的容量：所有正向割边的容量和**

**最小割：容量和最小的割**

**定理1：如果f是网络中的一个流，那么f的流为割的正向边与逆向边容量之差**

**定理二：如果f是一个流，CUT (S,T)是一个割，且f的值等于割CUT(S,T)的容量，那么f是一个最大流， CUT(S,T)是一个最小割。**

**推论：f的流量<=此流的割的容量**

**求最小割与求最大网络流等价**

POJ 1273 Drainage Ditches

**Description**

Every time it rains on Farmer John's fields, a pond forms over Bessie's favorite clover patch. This means that the clover is covered by water for awhile and takes quite a long time to regrow. Thus, Farmer John has built a set of drainage ditches so that Bessie's clover patch is never covered in water. Instead, the water is drained to a nearby stream. Being an ace engineer, Farmer John has also installed regulators at the beginning of each ditch, so he can control at what rate water flows into that ditch.   
Farmer John knows not only how many gallons of water each ditch can transport per minute but also the exact layout of the ditches, which feed out of the pond and into each other and stream in a potentially complex network.   
Given all this information, determine the maximum rate at which water can be transported out of the pond and into the stream. For any given ditch, water flows in only one direction, but there might be a way that water can flow in a circle.

**Input**

***The input includes several cases.*** For each case, the first line contains two space-separated integers, N (0 <= N <= 200) and M (2 <= M <= 200). N is the number of ditches that Farmer John has dug. M is the number of intersections points for those ditches. Intersection 1 is the pond. Intersection point M is the stream. Each of the following N lines contains three integers, Si, Ei, and Ci. Si and Ei (1 <= Si, Ei <= M) designate the intersections between which this ditch flows. Water will flow through this ditch from Si to Ei. Ci (0 <= Ci <= 10,000,000) is the maximum rate at which water will flow through the ditch.

**Output**

For each case, output a single integer, the maximum rate at which water may emptied from the pond.

**Sample Input**

5 4

1 2 40

1 4 20

2 4 20

2 3 30

3 4 10

**Sample Output**

50

**题意**

现在有m个池塘(从1到m开始编号,1为源点,m为汇点),及n条水渠,给出这n条水渠所连接的点和所能流过的最大流量，

**HDU 4183 Pahom on Water**

**Problem Description**

Pahom on Water is an interactive computer game inspired by a short story of Leo Tolstoy about a poor man who, in his lust for land, forfeits everything. The game's starting screen displays a number of circular pads painted with colours from the visible light spectrum. More than one pad may be painted with the same colour (defined by a certain frequency) except for the two colours red and violet. The display contains only one red pad (the lowest frequency of 400 THz) and one violet pad (the highest frequency of 789 THz). A pad may intersect, or even contain another pad with a different colour but never merely touch its boundary. The display also shows a figure representing Pahom standing on the red pad.  
The game's objective is to walk the figure of Pahom from the red pad to the violet pad and return back to the red pad. The walk must observe the following rules:  
1.If pad α and pad β have a common intersection and the frequency of the colour of pad α is strictly smaller than the frequency of the colour of pad β, then Pahom figure can walk from α to β during the walk from the red pad to the violet pad  
2. If pad α and pad β have a common intersection and the frequency of the colour of pad α is strictly greater than the frequency of the colour of pad β, then Pahom figure can walk from α to β during the walk from the violet pad to the red pad  
3. A coloured pad, with the exception of the red pad, disappears from display when the Pahom figure walks away from it.  
The developer of the game has programmed all the whizzbang features of the game. All that is left is to ensure that Pahom has a chance to succeed in each instance of the game (that is, there is at least one valid walk from the red pad to the violet pad and then back again to the red pad.) Your task is to write a program to check whether at least one valid path exists in each instance of the game.

**Input**

The input starts with an integer K (1 <= K <= 50) indicating the number of scenarios on a line by itself. The description for each scenario starts with an integer N (2 <= N <= 300) indicating the number of pads, on a line by itself, followed by N lines that describe the colors, locations and sizes of the N pads. Each line contains the frequency, followed by the x- and y-coordinates of the pad's center and then the radius. The frequency is given as a real value with no more than three decimal places. The coordinates and radius are given, in meters, as integers. All values are separated by a single space. All integer values are in the range of -10,000 to 10,000 inclusive. In each scenario, all frequencies are in the range of 400.0 to 789.0 inclusive. Exactly one pad will have a frequency of “400.0” and exactly one pad will have a frequency of “789.0”.

**Output**

The output for each scenario consists of a single line that contains: Game is VALID, or Game is NOT VALID

**Sample Input**

2

2

400.0 0 0 4

789.0 7 0 2

4

400.0 0 0 4

789.0 7 0 2

500.35 5 0 2

500.32 5 0 3

**Sample Output**

Game is NOT VALID

Game is VALID

**题目大意：**

有多个点，每个点给出坐标与半径，加入两个点相交，就可以从这两个点走。题目要求先从起点到终点，再从终点回到起点。从起点到终点的过程中，只能从频率小的走到频率大的点（前提是两点相交），从终点到起点的过程中，只能从频率大的走到频率小的。在走的过程中，除了起点与终点，别的只要走过就会消失，就是说只能走一次。问可不可以从起点到终点又回到起点。

**找到输入数据中最小和最大的两个，求两个之间的最大网络流**