

图论专题训练

写在前面的话：

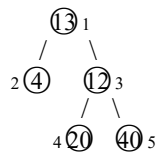
- 1、 所有题目的时间限制：1s，空间限制：128M；
- 2、 做题时间：210min；
- 3、 本套题目的目的：
 - 1) 锻炼大家 DP 能力，做题策略如此多的情况，看下能否找出最优决策，得到最高分；
 - 2) 顺带将自己熟悉的算法巩固下；
 - 3) 锻炼大家的编码能力；
 - 4) 找出自己图论薄弱的环节。
- 4、 温馨提示：
 - 1) 题目顺序与难度无关；
 - 2) 图论专题训练是否都是图论的题目呢？我也不知道。
- 5、 预祝大家取得高分`(*∩_∩*)`

1、医院设置

源程序名	hospital.??? (pas, c, cpp)
可执行文件名	hospital.exe
输入文件名	hospital.in
输出文件名	hospital.out

【问题描述】

设有一棵二叉树，如图



其中，圈中的数字表示结点中居民的人口。圈边上数字表示结点编号，现在要求在某个结点上建立一个医院，使所有居民所走的路程之和为最小，同时约定，相邻接点之间的距离为1。如上图中，若医院建在：3

【输入】

第一行一个整数 n ，表示树的结点数。($n \leq 100$)

接下来的 n 行每行描述了一个结点的状况，包含三个整数，整数之间用空格(一个或多个)分隔，其中：第一个数为居民人口数；第二个数为左链接，为0表示无链接；第三个数为右链接。

【输出】

一个整数，表示最小距离和。

【样例】

hospital.in	hospital.out
5	81
13 2 3	
4 0 0	
12 4 5	
20 0 0	
40 0 0	

2、工程规划

源程序名	work.??? (pas, c, cpp)
可执行文件名	work.exe
输入文件名	work.in
输出文件名	work.out

【问题描述】

造一幢大楼是一项艰巨的工程，它是由 n 个子任务构成的，给它们分别编号 $1, 2, \dots, n$ ($5 \leq n \leq 1000$)。由于对一些任务的起始条件有着严格的限制，所以每个任务的起始时间 T_1, T_2, \dots, T_n 并不是很容易确定的(但这些起始时间都是非负整数，因为它们必须在整个工程

开始后启动)。例如：挖掘完成后，紧接着就要打地基；但是混凝土浇筑完成后，却要等待一段时间再去掉模板。

这种要求就可以用 $M(5 \leq m \leq 5000)$ 个不等式表示，不等式形如 $T_i - T_j \leq b$ 代表 i 和 j 的起始时间必须满足的条件。每个不等式的右边都是一个常数 b ，这些常数可能不相同，但是它们都在区间 $(-100, 100)$ 内。

你的任务就是写一个程序，给定像上面那样的不等式，找出一种可能的起始时间序列 T_1, T_2, \dots, T_n ，或者判断问题无解。对于有解的情况，要使最早进行的那个任务和整个工程的起始时间相同，也就是说， T_1, T_2, \dots, T_n 中至少有一个为 0。

【输入】

第一行是用空格隔开的两个正整数 n 和 m ，下面的 m 行每行有三个用空格隔开的整数 i, j, b 对应着不等式 $T_i - T_j \leq b$ 。

【输出】

如果有可行的方案，那么输出 N 行，每行都有一个非负整数且至少有一个为 0，按顺序表示每个任务的起始时间。如果没有可行的方案，就输出信息 “NO SOLUTION”。

【样例 1】

work.in	work.out
5 8	0
1 2 0	2
1 5 -1	5
2 5 1	4
3 1 5	1
4 1 4	
4 3 -1	
5 3 -1	
5 4 -3	

【样例 2】

work.in	work.out
5 5	NO SOLUTION
1 2 -3	
1 5 -1	
2 5 -1	
5 1 -5	
4 1 4	

3、公路修建

源程序名	road.??? (pas, c, cpp)
可执行文件名	road.exe
输入文件名	road.in
输出文件名	road.out

【问题描述】

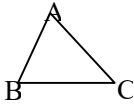
某国有 n 个城市，它们互相之间没有公路相通，因此交通十分不便。为解决这一“行路难”的问题，政府决定修建公路。修建公路的任务由各城市共同完成。

修建工程分若干轮完成。在每一轮中，每个城市选择一个与它最近的城市，申请修建通

往该城市的公路。政府负责审批这些申请以决定是否同意修建。

政府审批的规则如下：

- (1) 如果两个或以上城市申请修建同一条公路，则让它们共同修建；
- (2) 如果三个或以上的城市申请修建的公路成环。如下图，A 申请修建公路 AB，B 申请修建公路 BC，C 申请修建公路 CA。则政府将否决其中最短的一条公路的修建申请；



- (3) 其他情况的申请一律同意。
- 一轮修建结束后，可能会有若干城市可以通过公路直接或间接相连。这些可以互相：连通的城市即组成“城市联盟”。在下一轮修建中，每个“城市联盟”将被看作一个城市，发挥一个城市的作用。
- 当所有城市被组合成一个“城市联盟”时，修建工程也就完成了。
- 你的任务是根据城市的分布和前面讲到的规则，计算出将要修建的公路总长度。

【输入】

第一行一个整数 n ，表示城市的数量。 $(n \leq 5000)$
以下 n 行，每行两个整数 x 和 y ，表示一个城市的坐标。 $(-1000000 \leq x, y \leq 1000000)$

【输出】

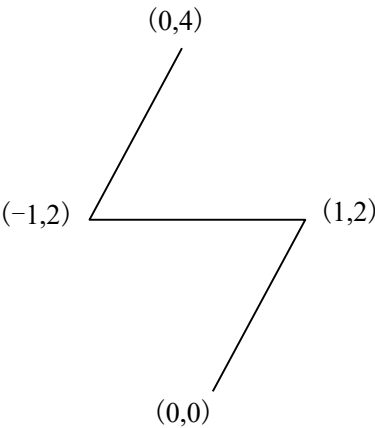
一个实数，四舍五入保留两位小数，表示公路总长。(保证有惟一解)

【样例】

road.in
4
0 0
1 2
-1 2
0 4

road.out
6.47

修建的公路如图所示：



4、K-联赛

源程序名	kleague.??? (pas, c, cpp)
可执行文件名	kleague.exe
输入文件名	kleague.in
输出文件名	kleague.out

【问题描述】

K-联赛职业足球俱乐部的球迷们都是有组织的训练有素的啦啦队员，就像红魔啦

啦啦队一样(2002 年韩日世界杯上韩国队的啦啦队)。这个赛季, 经过很多场比赛以后, 球迷们希望知道他们支持的球队是否还有机会赢得最后的联赛冠军。换句话说, 球队是否可以通过某种特定的比赛结果最终取得最高的积分(获胜场次最多)。(允许出现多支队并列第一的情况。)

现在, 给出每个队的胜负场数, w_i 和 d_i , 分别表示 $team_i$ 的胜场和负场 ($1 \leq i \leq n$)。还给出 a_{ij} , 表示 $team_i$ 和 $team_j$ 之间还剩多少场比赛要进行 ($1 \leq i, j \leq n$)。这里, n 表示参加联赛的队数, 所有的队分别用 $1, 2, \dots, n$ 来编号。你的任务是找出所有还有可能获得冠军的球队。

所有队参加的比赛数是相同的, 并且为了简化问题, 你可以认为不存在平局(比赛结果只有胜或负两种)。

【输入】

第一行一个整数 n ($1 \leq n \leq 25$), 表示联赛中的队数。

第二行 $2n$ 个数, $w_1, d_1, w_2, d_2, \dots, w_n, d_n$, 所有的数不超过 100。

第三行 n^2 个数, $a_{1,1}, a_{1,2}, \dots, a_{1,n}, a_{2,1}, \dots, a_{2,2}, a_{2,n}, \dots, a_{n,1}, a_{n,2}, \dots, a_{n,n}$, 所有的数都不超过 10。 $a_{i,j}=a_{j,i}$, 如果 $i=j$, 则 $a_{i,j}=0$ 。

【输出】

仅一行, 输出所有可能获得冠军的球队, 按其编号升序输出, 中间用空格分隔。

【样例 1】

kleague.in	kleague.out
3	1 2 3
2 0 1 1 0 2	
0 2 2 2 0 2 2 0	

【样例 2】

kleague.in	kleague.out
3	1 2
4 0 2 2 0 4	
0 1 1 1 0 1 1 1 0	

【样例 3】

kleague.in	kleague.out
4	2 4
0 3 3 1 1 3 3 0	
0 0 0 2 0 0 1 0 0 1 0 0 2 0 0 0	

5、速度限制

源程序名	speed.??? (pas, c, cpp)
可执行文件名	speed.exe
输入文件名	speed.in
输出文件名	speed.out

【问题描述】

在这个繁忙的社会中, 我们往往不再去选择最短的道路, 而是选择最快的路线。开车时每条道路的限速成为最关键的问题。不幸的是, 有一些限速的标志丢失了, 因此你无法得知应该开多快。一种可以辩解的解决方案是, 按照原来的速度行驶。你的任务是计算两地间的最快路线。

你将获得一份现代化城市的道路交通信息。为了使问题简化, 地图只包括路口和道路。每条道路是有向的, 只连接了两条道路, 并且最多只有一块限速标志, 位于路的起点。两地

A 和 B，最多只有一条道路从 A 连接到 B。你可以假设加速能够在瞬间完成并且不会有交通堵塞等情况影响你。当然，你的车速不能超过当前的速度限制。

【输入】

输入文件 SPEED.IN 的第一行是 3 个整数 N, M 和 D ($2 \leq N \leq 150$)，表示道路的数目，用 0..N-1 标记。M 是道路的总数，D 表示你的目的地。接下来的 M 行，每行描述一条道路，每行有 4 个整数 A ($0 \leq A < N$)，B ($0 \leq B < N$)，V ($0 \leq V \leq 500$) and L ($1 \leq L \leq 500$)，这条路是从 A 到 B 的，速度限制是 V，长度为 L。如果 V 是 0，表示这条路的限速未知。如果 V 不为 0，则经过该路的时间 $T=L/V$ 。否则 $T=L/V_{old}$ ， V_{old} 是你到达该路口前的速度。开始时你位于 0 点，并且速度为 70。

【输出】

输出文件 SPEED.OUT 仅一行整数，表示从 0 到 D 经过的城市。

输出的顺序必须按照你经过这些城市的顺序，以 0 开始，以 D 结束。仅有一条最快路线。

【样例】

speed.in	speed.out
6 15 1	0 5 2 3 1
0 1 25 68	
0 2 30 50	
0 5 0 101	
1 2 70 77	
1 3 35 42	
2 0 0 22	
2 1 40 86	
2 3 0 23	
2 4 45 40	
3 1 64 14	
3 5 0 23	
4 1 95 8	
5 1 0 84	
5 2 90 64	
5 3 36 40	

【问题分析】

首先，利用预处理计算任意两个节点之间只经过无限速标志的路的最短距离。这可以用 Floyd 算法得到，时间复杂度为 $O(n^3)$ 。

计算城市 1 到城市 D 之间最快路径时，只需对 Dijkstra 稍作修改即可：在 Dijkstra 算法中，用一个已计算出最短路径的节点去刷新其他节点当前最短路径长度时，除了要枚举有限速标志的路以外，还要在此路的基础上，枚举通过此路后要经过无限速标志的路到达的节点。时间复杂度为 $O(n^2+mn)$ ，即 $O(mn)$ 。

6、机器调度

源程序名	machine.??? (pas, c, cpp)
可执行文件名	machine.exe
输入文件名	machine.in
输出文件名	machine.out

【问题描述】

我们知道机器调度是计算机科学中一个非常经典的问题。调度问题有很多种，具体条件不同，问题就不同。现在我们要处理的是两个机器的调度问题。

有两个机器 A 和 B。机器 A 有 n 种工作模式，我们称之为 $mode_0, mode_1, \dots, mode_n-1$ 。同样，机器 B 有 m 种工作模式，我们称之为 $mode_0, mode_1, \dots, mode_m-1$ 。初始时，两台机器的工作模式均为 $mode_0$ 。现在有 k 个任务，每个工作都可以在两台机器中任意一台的特定的模式下被加工。例如， job_0 能在机器 A 的 $mode_3$ 或机器 B 的 $mode_4$ 下被加工， job_1 能在机器 A 的 $mode_2$ 或机器 B 的 $mode_4$ 下被加工，等等。因此，对于任意的 job_i ，我们可以用三元组 (i, x, y) 来表示 job_i 在机器 A 的 $mode_x$ 或机器 B 的 $mode_y$ 下被加工。

显然，要完成所有工作，我们需要不时的改变机器的工作模式。但是，改变机器的工作状态就必须重启机器，这是需要代价的。你的任务是，合理的分配任务给适当的机器，使机器的重启次数尽量少。

【输入】

第一行三个整数 n, m ($n, m < 100$), k ($k < 1000$)。接下来的 k 行，每行三个整数 i, x, y 。

【输出】

只一行一个整数，表示最少的重启次数。

【样例】

machine.in	machine.out
5 5 10	3
0 1 1	
1 1 2	
2 1 3	
3 1 4	
4 2 1	
5 2 2	
6 2 3	
7 2 4	
8 3 3	
9 4 3	

【问题分析】

本题所求的是工作模式的最少切换次数，实际上也就是求最少需要使用多少个工作模式，因为一个工作模式被切换两次肯定是不合算的，一旦切换到一个工作模式就应该把这个工作模式可以完成的工作都完成。

将两台机器的工作模式分别看成 n 个和 m 个节点。 job_i 分别和机器 A 和 B 的 $mode_x$ 和 $mode_y$ 相关： job_i 要被完成，就必须切换到机器 A 的 $mode_x$ 或切换到机器 B 的 $mode_y$ 。将 job_i 看作图中的一条边——连接节点 x 和节点 y 的边，那么这条边就要求 x 和 y 两个节点中至少要有有一个节点被取出来。这正符合覆盖集的性质。

我们构成的图是二分图，要求最少的切换次数，就是要使覆盖集最小。二分图的最小覆盖集问题等价于二分图的最大匹配问题。因此，只需对此二分图求一个最大匹配即是我们要求的答案。时间复杂度 $O((m+n)^{\frac{1}{2}}k)$ 。

7、邮递员送信

源程序名	post.??? (pas, c, cpp)
可执行文件名	post.exe
输入文件名	post.in

输出文件名	post.out
-------	----------

【问题描述】

有一个邮递员要送东西，邮局在节点 1.他总共要送 $N-1$ 样东西，其目的地分别是 $2\sim N$ 。由于这个城市的交通比较繁忙，因此所有的道路都是单行的，共有 M 条道路，通过每条道路需要一定的时间。这个邮递员每次只能带一样东西。求送完这 $N-1$ 样东西并且最终回到邮局最少需要多少时间。

【输入文件】

输入文件第一行包括一个正整数 N 和 M ；

接下来 M 行，每行三个正整数 U,V,W ,表示该条道路为从 U 到 V 的，且通过这条道路需要 W 的时间。满足 $1\leq U,V\leq N, 1\leq W\leq 10000$,输入保证任意两点都能互相到达。

【输出文件】

输出仅一行，包含一个整数，为最少需要的时间。

【样例输入】

```
5 10
2 3 5
1 5 5
3 5 6
1 2 8
1 3 8
5 3 4
4 1 8
4 5 3
3 5 6
5 4 2
```

【样例输出】

```
83
```

8、拐弯

源程序名	ddos.??? (pas, c, cpp)
可执行文件名	ddos.exe
输入文件名	ddos.in
输出文件名	ddos.out

【题目描述】

$N*N$ ($1\leq N\leq 10$) 方格中，‘x’表示不能行走的格子，‘.’表示可以行走的格子。卡门很胖，故而不好转弯。现在要从 A 到走到 B 点，请问最少要转 90 度弯几次？

【输入格式】

第一行一个整数：N，下面 N 行，每行 N 个字符，只出现字符：‘.’，‘x’，‘A’，‘B’；

表示上面所说的矩阵格子，每个字符后有一个空格。

【输出格式】

一个整数：最少转弯次数。如果不能到达，输出-1。

【样例输入】

3
.xA
...
Bx.

【样例输出】

2

【数据范围】

30%的数据 $2 \leq N \leq 5$

50%的数据 $2 \leq N \leq 30$

100%的数据 $2 \leq N \leq 100$

【注释】

只可以上、下、左、右四个方向行走，并且不能走出这些格子之外。开始和结束时的方向可以任意。