**CCF全国信息学奥林匹克联赛（NOIP2014）复赛**

提高组 day2

（请选手务必仔细阅读本页内容）

**一．题目概况**

|  |  |  |  |
| --- | --- | --- | --- |
| 中文题目名称 | 无线网路发射器选址 | 寻找道路 | 解方程 |
| 英文题目与子目录名 | wireless | road | equation |
| 可执行文件名 | wireless | road | equation |
| 输入文件名 | wireless.in | road.in | equation.in |
| 输出文件名 | wireless.out | road.out | equation.out |
| 每个测试点时限 | 1秒 | 1秒 | 1秒 |
| 测试点数目 | 10 | 10 | 20 |
| 每个测试点分值 | 10 | 10 | 5 |
| 附加样例文件 | 有 | 有 | 有 |
| 结果比较方式 | 全文比较（过滤行末空格及文末回车） | | |
| 题目类型 | 传统 | 传统 | 传统 |
| 运行内存上限 | 128M | 128M | 128M |

**二．ᨀ交源程序文件名**

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++语言 | wireless.cpp | road.cpp | equation.cpp |
| 对于C语言 | wireless.c | road.c | equation.c |
| 对于pascal语言 | wireless.pas | road.pas | equation.pas |

**三．编译命令（不包含任何优化开关）**

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++语言 | g++ -o wireless wireless.cpp–lm | g++ -o road road.cpp –lm | g++ -o equation  equation.cpp–lm |
| 对于C语言 | gcc -o wireless  wireless.c –lm | gcc -o road road.c –lm | gcc -o equation  equation.c –lm |
| 对于pascal语言 | fpc wireless.pas | fpc road.pas | fpc equation.pas |

**注意事项：**

1、文件名（程序名和输入输出文件名）必须使用英文小写。

2、C/C++中函数main()的返回值类型必须是int，程序正常结束时的返回值必须是0。

3、全国统一评测时采用的机器配置为：CPU AMD Athlon(tm) 64x2 Dual Core CPU 5200+，2.71GHz，内存2G，上述时限以此配置为准。

4、只ᨀ供Linux格式附加样例文件。

**5、特别提醒 ：评测在当前最新公布的NOI Linux下进行，各语言的编译器版本以其为准。**

**1．无线网络发射器选址**

**(wireless.cpp/c/pas)**

**【问题描述】**

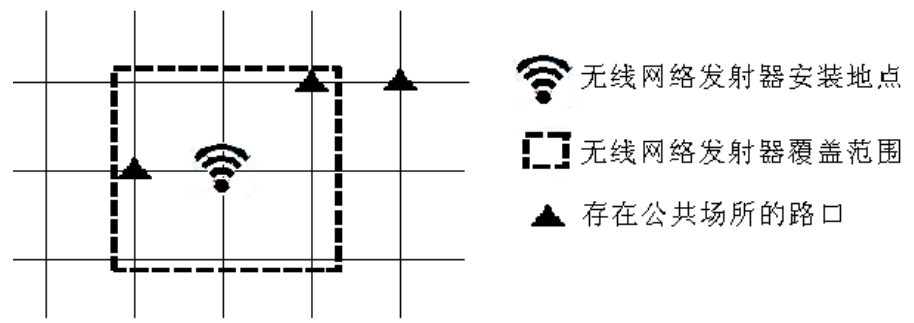
随着智能手机的日益普及，人们对无线网的需求日益增大。某城市决定对城市内的公共场所覆盖无线网。

假设该城市的布局为由严格平行的129条东西向街道和129条南北向街道所形成的网格状，并且相邻的平行街道之间的距离都是恒定值1。东西向街道从北到南依次编号为0,1,2…128,南北向街道从西到东依次编号为0,1,2…128。

东西向街道和南北向街道相交形成路口，规定编号为x的南北向街道和编号为y的东西向街道形成的路口的坐标是（x, y）。 在 某 些 路 口 存 在 一 定 数 量 的 公 共 场 所 。

由于政府财政问题，只能安装一个大型无线网络发射器。该无线网络发射器的传播范围是一个以该点为中心，边长为2\*d的正方形。传播范围包括正方形边界。

例如下图是一个d = 1的无线网络发射器的覆盖范围示意图。



现在政府有关部门准备安装一个传播参数为d的无线网络发射器，希望你帮助他们在城市内找出合适的安装地点，使得覆盖的公共场所最多。

**【输入】**

输入文件名为wireless.in。

第一行包含一个整数d，表示无线网络发射器的传播距离。

第二行包含一个整数n，表示有公共场所的路口数目。

接下来n行，每行给出三个整数x, y, k, 中间用一个空格隔开，分别代表路口的坐标(x, y)以及该路口公共场所的数量。同一坐标只会给出一次。

**【输出】**

输出文件名为wireless.out。

输出一行，包含两个整数，用一个空格隔开，分别表示能覆盖最多公共场所的安装地点方案数，以及能覆盖的最多公共场所的数量。

**【输入输出样例】**

|  |  |
| --- | --- |
| **wireless.in** | **wireless.out** |
| 1  2  4 4 10  6 6 20 | 1 30 |

**【数据说明】**

对于100%的数据，1 ≤ d ≤ 20，1 ≤ n ≤ 20， 0 ≤ x ≤ 128, 0 ≤ y ≤ 128, 0 < k ≤ 1,000,000。

**2．寻找道路**

**(road.cpp/c/pas)**

**【问题描述】**

在有向图G中，每条边的长度均为1，现给定起点和终点，请你在图中找一条从起点到终点的路径，该路径满足以下条件：

1．路径上的所有点的出边所指向的点都直接或间接与终点连通。

2．在满足条件1的情况下使路径最短。

注意：图G中可能存在重边和自环，题目保证终点没有出边。

请你输出符合条件的路径的长度。

**【输入】**

输入文件名为road.in。

第一行有两个用一个空格隔开的整数n和m，表示图有n个点和m条边。

接下来的m行每行2个整数x、y，之间用一个空格隔开，表示有一条边从点x指向点y。

最后一行有两个用一个空格隔开的整数s、t，表示起点为s，终点为t。

**【输出】**

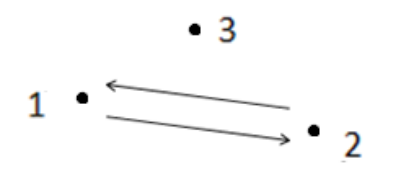
输出文件名为road.out。

输出只有一行，包含一个整数，表示满足题目描述的最短路径的长度。如果这样的路径不存在，输出-1。

【输入输出样例1】

|  |  |
| --- | --- |
| **road.in** | **road.out** |
| 3 2  1 2  2 1  1 3 | －1 |

**【输入输出样例说明】**

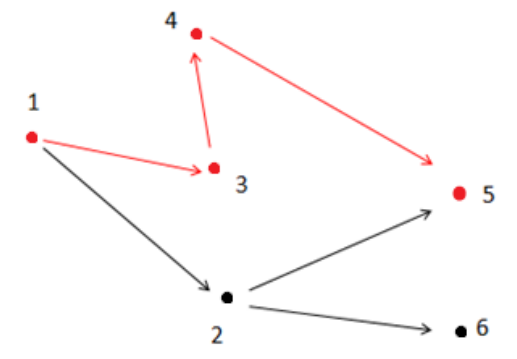


如上图所示，箭头表示有向道路，圆点表示城市。起点1与终点3不连通，所以满足题目描述的路径不存在，故输出-1。

【输入输出样例2】

|  |  |
| --- | --- |
| **road.in** | **road.out** |
| 6 6  1 2  1 3  2 6  2 5  4 5  3 4  1 5 | 3 |

**【输入输出样例说明】**



如上图所示，满足条件的路径为1->3->4->5。注意点2不能在答案路径中，因为点2连了一条边到点6，而点6不与终点5连通。

【数据说明】

对于30%的数据，0< n ≤10，0< m ≤20；

对于60%的数据，0< n ≤100，0< m ≤2000；

对于100%的数据，0< n ≤10,000，0< m ≤200,000，0< x,y,s,t≤n，x≠t。

**3．解方程**

**(equation.cpp/c/pas)**

**【问题描述】**

已知多项式方程：



求这个方程在[1, m]内的整数解（n和m均为正整数）。

**【输入】**

输入文件名为equation.in。

输入共n+2行。

第一行包含2个整数n、m，每两个整数之间用一个空格隔开。

接下来的n+1行每行包含一个整数，依次为a0,a1,a2,……,an。

【输出】

输出文件名为equation.out。

第一行输出方程在[1, m]内的整数解的个数。

接下来每行一个整数，按照从小到大的顺序依次输出方程在[1, m]内的一个整数解。

【输入输出样例1】

|  |  |
| --- | --- |
| **equation.in** | **equation.out** |
| 2 10  1  -2  1 | 1  1 |

**【输入输出样例2】**

|  |  |
| --- | --- |
| **equation.in** | **equation.out** |
| 2 10  2  -3  1 | 2  1  2 |

**【输入输出样例3】**

|  |  |
| --- | --- |
| **equation.in** | **equation.out** |
| 2 10  1  3  2 | 0 |

**【数据说明】**

对于30%的数据，0<n≤2，|ai|≤100，an≠0，m≤100；

对于50%的数据，0<n≤100，|ai|≤10100，an≠0，m≤100；

对于70%的数据，0<n≤100，|ai|≤1010000，an≠0，m≤10000；

对于100%的数据，0<n≤100，|ai|≤1010000，an≠0，m≤1000000。