

关系挖掘

【问题描述】

有人说：“世界上任何两个人最多只需要通过 6 个人就能建立联系”。小 B 对此非常感兴趣，于是着手开展了一些社会网络中关系挖掘的研究工作。

现在小 B 得到了一个包含 N 个人的社会网络数据，该网络中有 M 条关系信息。一条关系信息可以表示为 (a_i, b_i, w_i) ，表示 a_i 与 b_i 两个人之间存在关系，且紧密度为 w_i ($w_i > 0$)。现在小 B 希望挑选其中的 K ($K \leq N$) 个人作为研究对象，为使研究工作具有较高的可信度，小 B 希望这 K 个人之间的关系紧密度之和尽量大。

该问题可抽象为：给定一个带权无向图 $G=(V, E)$ 和整数 K ，目标是求出点集 V 的一个子集 S ，使得 $|S| = K$ ，且使下式最大化

$$\sum_{(a_i, b_i, w_i) \in E \text{ 且 } a_i \in S \text{ 且 } b_i \in S} w_i$$

【输入格式】

这是一道提交答案的试题，在你的目录下有 10 个输入文件 `relation*.in`。

输入文件的第一行为三个整数 N ， M 和 K ，分别表示给定的社会网络中的点数（人数）、边数（关系数目）以及需要选择的点数（人数） K 。

接下来 M 行，每行三个正整数 a_i ， b_i ， w_i ，表示一条边（关系）。所有的点（人）按 1 到 N 依次编号。

【输出格式】

对于每一个输入文件，在目录下给出对应的输出文件 `relation*.out`。

输出文件应包含 K 行，每行一个整数，表示选出的 K 个人的编号。

【样例输入】

```
3 2 2
1 2 3
1 3 5
```

【样例输出】

```
1
3
```

【评分标准】

对于每个测试点，如果你没有输出或者输出不合法则得 0 分。

对于每个测试点，我们设有四个评分参数 m_1 、 m_2 、 m_3 和 m_4 。假设选手选出的 K 个人之间关系紧密度之和为 c ，

若 $c > m_1$ ，得 12 分；

若 $c = m_1$ ，得 10 分；

若 $m_1 > c \geq m_2$ ，得 8 分；

若 $m_2 > c \geq m_3$ ，得 5 分；

若 $m_3 > c \geq m_4$ ，得 3 分；

若 $c > 0$ ，得 1 分；

否则，得 0 分。

【如何测试你的输出】

在你的目录下有一个名为 `checker` 的程序可以用来检查你的输出，你可以在终端中使用以下命令来检查你的输出：

```
./checker N
```

其中 N 为测试点的编号，例如，要测试第 3 个测试点可以使用

```
./checker 3
```

该程序会检测你的输出是否合法。如果方案合法，程序还会给出该方案的紧密度之和。