

最小算法生成树 — 在城市交通规划中的应用

1. 问题描述

某地区经过对城镇交通状况的调查，得到现有城镇间快速道路的统计数据，并提出“畅通工程”的目标：使整个地区任何两个城镇间都可以实现快速交通（不一定有直接的快速道路相连，只要互相间接通过快速路可达即可）。现得到城镇道路统计表，表中列出了有可能建设成快速路的若干条道路的成本，求畅通工程需要的最低成本。

1. 问题输入

- 第一行给出城镇数目 N ($1 < N \leq 1000$) 和候选道路数目 $M \leq 3N$
- 随后的 M 行，每行给出 3 个正整数，分别是该条道路直接连通的两个城镇的编号（1 ~ N ）以及该道路修建的预算成本

2. 问题输出

- 输出“畅通工程”需要的最低成本，如果问题输入数据无法保证畅通，则输出“警告：输入数据无法满足道路畅通”
- 如果可以实现“畅通工程”的目标，则输出道路畅通图的邻接矩阵。

2. 问题分析

“畅通工程”问题可以看作是图论中的最小耗费生成树（Minimum Cost Spanning Tree）问题，其中城镇是图中的节点，候选道路是图中的边，而每条边上的成本则代表边的权重。其目标是找到一棵生成树，使得所有城镇都能通过这些道路连通，且总成本最小。

解决该问题的步骤如下：

- 构建图：**将城镇和候选道路构建成图 $G(V, E)$ ，城镇为节点，候选道路为边，边的权重为修建成本。
- 最小生成树算法：**使用最小生成树算法（如 Prim 算法或 算法）找到满足条件的最小生成树。
- 检查是否畅通：**检查生成树是否包含所有的城镇。如果生成树包含所有城镇，则畅通；否则，无法畅通，输出警告信息。
- 计算最低成本：**如果畅通，则计算生成树的总成本即为所求的最低成本。

3. 所选算法详述 — Kruskal's Algorithm

Kruskal 算法是一种用于在带权边的连通无向图中找到最小生成树的贪婪算法，其目标是找到一个包含图所有顶点且边权和最小的生成树。

以下是Kruskal算法的详细步骤：

- 1. 初始化：** 将图中的所有边按照权值从小到大进行排序。
- 2. 创建一个空的最小生成树：** 初始时，最小生成树不包含任何边。
- 3. 遍历排序后的边集：** 从排好序的边集中逐一选择边，每次选择权值最小的边。
- 4. 检查边的两个顶点是否在同一连通分量中：** 使用并查集（Union-Find）数据结构来判断选择的边的两个顶点是否已经属于同一个连通分量。如果不在同一连通分量中，将这条边添加到最小生成树中，并将这两个顶点合并为一个连通分量。
- 5. 重复步骤 3 和步骤 4：** 重复这个过程，直到最小生成树中包含了所有的顶点。

4. 实验结果与分析

- 输入

```
6 9
1 2 1
1 3 2
2 3 6
2 4 11
3 4 9
3 5 13
4 5 7
4 6 3
5 6 4
```

- 输出

```
>>>道路畅通的最小成本为 19
>>>最小成本的通路对应的邻接矩阵为
inf 1 2 inf inf inf
1 inf inf inf inf inf
2 inf inf 9 inf inf
inf inf 9 inf inf 3
inf inf inf inf inf 4
inf inf inf 3 4 inf
```

