

{% note info %} 摘要 Title: 859. Kruskal算法求最小生成树 Tag: kruskal、最小生成树 Memory Limit: 64 MB Time Limit: 1000 ms {% endnote %}

Powered by: NEFU AB-IN

[Link](#)

@TOC

859. Kruskal算法求最小生成树

- 题意

给定一个 n 个点 m 条边的无向图，图中可能存在重边和自环，边权可能为负数。求最小生成树的树边权重之和，如果最小生成树不存在则输出 impossible。给定一张边带权的无向图 $G=(V,E)$ ，其中 V 表示图中点的集合， E 表示图中边的集合， $n=|V|$ ， $m=|E|$ 。由 V 中的全部 n 个顶点和 E 中 $n-1$ 条边构成的无向连通子图被称为 G 的一棵生成树，其中边的权值之和最小的生成树被称为无向图 G 的最小生成树。

- 思路

Kruskal板子题

Kruskal 算法

① 将所有边按权重从小到大排序 $O(m \log m)$

57164

② 枚举每条边 a, b 权重 c

if a, b 不连通

将这条边加入集合中

$O(m)$

- 代码

```
...
Author: NEFU AB-IN
Date: 2022-03-03 19:43:34
FilePath: \ACM\Acwing\859.py
LastEditTime: 2022-03-03 19:57:59
...

N = int(2e5 + 100)
fa = [i for i in range(N)]

def find(x):
    if fa[x] != x:
        fa[x] = find(fa[x])
    return fa[x]

class Edge(object):
    def __init__(self, u, v, w):
        self.u = u
        self.v = v
        self.w = w

    def __lt__(self, t):
        return self.w < t.w

lst = []
n, m = map(int, input().split())
for i in range(m):
    x, y, z = map(int, input().split())
    lst.append(Edge(x, y, z))
lst.sort()

res, cnt = 0, 0
for i in range(m):
    u, v, w = lst[i].u, lst[i].v, lst[i].w
    u = find(u)
    v = find(v)
    if u != v:
        fa[u] = v
        cnt += 1
        res += w

if cnt < n - 1: #当边的数量小于n-1时
    print("impossible")
else:
    print(res)
```