



Contest Menu

CONTESTANT

Overview

Announcements

Problems

Submissions

Clarifications

Scoreboard

APIO 2020 CHN Mirror

Contest is over.

Chinese (zh-Hans-CN)

Switch

B. Swapping Cities

Time limit	2 s
Memory limit	512 MB

题目描述

印度尼西亚有 N 个城市以及 M 条双向道路，城市从 0 到 $N - 1$ 编号，道路从 0 到 $M - 1$ 编号。每条道路连接着两个不同的城市，第 i 条道路连接第 $U[i]$ 个城市与第 $V[i]$ 个城市，汽车行驶这条道路将耗费 $W[i]$ 个单位汽油。通过这些道路，任意两个城市间能够互相到达。

接下来的 Q 天中，每天会有一对城市希望建立政治关系。具体来说，第 j 天，第 $X[j]$ 个城市想要和第 $Y[j]$ 个城市建立政治关系。为此，第 $X[j]$ 个城市将会派一名代表坐汽车前往第 $Y[j]$ 个城市。同样地，第 $Y[j]$ 个城市也会派一名代表坐汽车前往第 $X[j]$ 个城市。

为了避免拥塞，两辆车不应在任何时间点碰面。更具体地，两辆车不能在同一个时间点出现在同一个城市。同样地，两辆车也不应该沿相反的方向同时行驶过同一条道路。另外，汽车行驶过一条道路时必须完整经过道路并到达道路另一端的城市（换句话说，汽车不允许在道路中间掉转方向）。但是，汽车可以多次到达一个城市或是多次经过一条道路。此外，汽车可以在任何时间在任何城市等候。

由于高燃料容量汽车的价格昂贵，两个城市都分别希望选择一条路线，使得两辆汽车所需的最大单位汽油容量最小。每个城市中都有加油站并且供油量是无限的，因此汽车所需的单位汽油容量实际上就是行驶过的道路中最大的单位汽油消耗量。

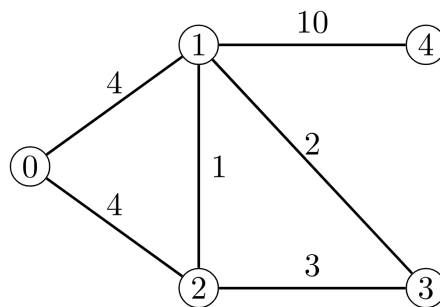
具体实现

你必须实现 `init` 和 `getMinimumFuelCapacity` 函数。

- `init(N, M, U, V, W)` - 该函数将在所有 `getMinimumFuelCapacity` 的调用前被评测库恰好调用一次。
 - N : 一个整数表示城市数。
 - M : 一个整数表示道路数。
 - U : 一个长为 M 的整数序列表示道路的第一个端点城市。
 - V : 一个长为 M 的整数序列表示道路的第二个端点城市。
 - W : 一个长为 M 的整数序列表示道路的汽油消耗。
- `getMinimumFuelCapacity(X, Y)` - 该函数将被评测库调用恰好 Q 次。
 - X : 一个整数表示第一个城市。
 - Y : 一个整数表示第二个城市。
 - 该函数必须返回一个整数，表示根据题目描述中的规则，两辆分别从第 X 个城市与第 Y 个城市出发要到达彼此城市的车，它们的单位汽油容量最大值的最小值。若无法满足题目规则则返回 -1 。

样例

第一个样例中, $N = 5, M = 6, U = [0, 0, 1, 1, 1, 2], V = [1, 2, 2, 3, 4, 3], W = [4, 4, 1, 2, 10, 3], Q = 3, X = [1, 2, 0], Y = [2, 4, 1]$ 。如下图:



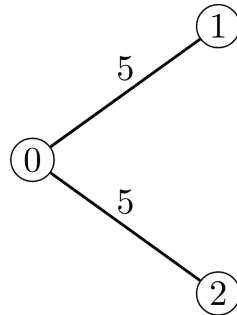
评测库初始时将调用 `init(5, 6, [0, 0, 1, 1, 1, 2], [1, 2, 2, 3, 4, 3], [4, 4, 1, 2, 10, 3])`。之后, 评测库将进行如下函数调用:

- `getMinimumFuelCapacity(1, 2)`。首先, 从第一个城市出发的汽车可以行驶到第三个城市。接着, 从第二个城市出发的汽车可以行驶到第一个城市, 并且在第三个城市的汽车可以行驶到第二个城市。因此, 最大的单位汽油容量为 3 (从第三个城市到第二个城市需要花费 3 个单位汽油)。没有其他更优的路线方案, 因此该函数应该返回 3。
- `getMinimumFuelCapacity(2, 4)`。任何从第四个城市出发或要到达第四个城市的汽车都需要耗费 10

个单位汽油，因此该函数应该返回 10。

- `getMinimumFuelCapacity(0, 1)`。该函数应该返回 4。

第二个样例中, $N = 3, M = 2, U = [0, 0], V = [1, 2], W = [5, 5], Q = 1, X = [1], Y = [2]$ 。如下图：



评测库初始时将调用 `init(3, 2, [0, 0], [1, 2], [5, 5])`。之后，评测库将进行如下函数调用：

- `getMinimumFuelCapacity(1, 2)`。两辆车无法满足不在同一时间点碰面的要求，所以该函数应该返回 -1。

条件限制

- $2 \leq N \leq 100\,000$.
- $N - 1 \leq M \leq 200\,000$.
- $0 \leq U[i] < V[i] < N$.
- 任意两个城市间至多存在一条道路直接相连。
- 任意两个城市经过道路可以互相到达。
- $1 \leq W[i] \leq 10^9$.
- $1 \leq Q \leq 200\,000$.
- $0 \leq X[j] < Y[j] < N$.

子任务 1 (6 分)

- 每个城市至多是两条道路的一个端点。

子任务 2 (7 分)

- $M = N - 1$.
- $U[i] = 0$.

子任务 3 (17 分)

- $Q \leq 5$.
- $N \leq 1\,000$.
- $M \leq 2\,000$.

子任务 4 (20 分)

- $Q \leq 5$.

子任务 5 (23 分)

- $M = N - 1$.

子任务 6 (27 分)

- 无附加条件。

样例评测库

样例评测库将读入以下格式的数据：

```
N M
U[0] V[0] W[0]
U[1] V[1] W[1]
.
.
.
U[M-1] V[M-1] W[M-1]
Q
X[0] Y[0]
X[1] Y[1]
.
.
.
X[Q-1] Y[Q-1]
```

对每个 `getMinimumFuelCapacity` 的调用，样例评测库会输出该函数的返回值。

附加文件

本题公开的评测库、样例测试点和程序框架可在此查看：[here](#).

Submit solution



Contest is over.