

4. 树网的核

(core.pas/c/cpp)

【问题描述】

设 $T=(V, E, W)$ 是一个无圈且连通的无向图（也称为无根树），每条边带有正整数的权，我们称 T 为树网（*treenetwork*），其中 V, E 分别表示结点与边的集合， W 表示各边长度的集合，并设 T 有 n 个结点。

路径：树网中任何两结点 a, b 都存在唯一的一条简单路径，用 $d(a, b)$ 表示以 a, b 为端点的路径的长度，它是该路径上各边长度之和。我们称 $d(a, b)$ 为 a, b 两结点间的距离。

一点 v 到一条路径 P 的距离为该点与 P 上的最近的结点的距离：

$$d(v, P) = \min\{d(v, u), u \text{ 为路径 } P \text{ 上的结点}\}.$$

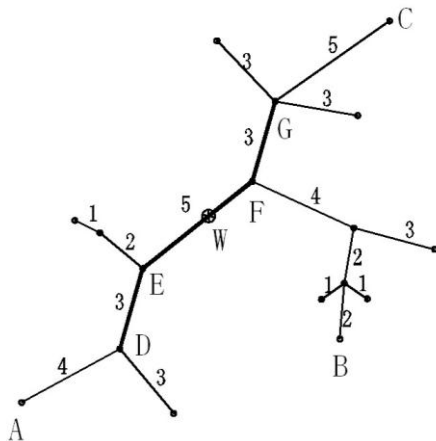
树网的直径：树网中最长的路径称为树网的直径。对于给定的树网 T ，直径不一定是唯一的，但可以证明：各直径的中点（不一定恰好是某个结点，可能在某条边的内部）是唯一的，我们称该点为树网的中心。

偏心距 $ECC(F)$ ：树网 T 中距路径 F 最远的结点到路径 F 的距离，即

$$ECC(F) = \max\{d(v, F), v \in V\}.$$

任务：对于给定的树网 $T=(V, E, W)$ 和非负整数 s ，求一个路径 F ，它是某直径上的一段路径（该路径两端均为树网中的结点），其长度不超过 s （可以等于 s ），使偏心距 $ECC(F)$ 最小。我们称这个路径为树网 $T=(V, E, W)$ 的核（*Core*）。必要时， F 可以退化为某个结点。一般来说，在上述定义下，核不一定只有一个，但最小偏心距是唯一的。

下面的图给出了树网的一个实例。图中，A-B 与 A-C 是两条直径，长度均为 20。点 W 是树网的中心，EF 边的长度为 5。如果指定 $s=11$ ，则树网的核为路径 DEFG（也可以取为路径 DEF），偏心距为 8。如果指定 $s=0$ （或 $s=1$ 、 $s=2$ ），则树网的核为结点 F，偏心距为 12。



【输入】

输入文件 core.in 包含 n 行：

第 1 行，两个正整数 n 和 s ，中间用一个空格隔开。其中 n 为树网结点的个数， s 为树网的核的长度的上界。设结点编号依次为 $1, 2, \dots, n$ 。

从第 2 行到第 n 行，每行给出 3 个用空格隔开的正整数，依次表示每一条边的两个端点编号和长度。例如，“2 4 7”表示连接结点 2 与 4 的边的长度为 7。

所给的数据都是正确的，不必检验。

【输出】

输出文件 `core.out` 只有一个非负整数，为指定意义下的最小偏心距。

【输入输出样例 1】

<code>core.in</code>	<code>core.out</code>
5 2 1 2 5 2 3 2 2 4 4 2 5 3	5

【输入输出样例 2】

<code>core.in</code>	<code>core.out</code>
8 6 1 3 2 2 3 2 3 4 6 4 5 3 4 6 4 4 7 2 7 8 3	5

【NOIP原版限制】

40%的数据满足： $5 \leq n \leq 15$

70%的数据满足： $5 \leq n \leq 80$

100%的数据满足： $5 \leq n \leq 300, 0 \leq s \leq 1000$ 。边长度为不超过 1000 的正整数

【BZOJ加强版限制】

对于70%的数据， $n \leq 200000$

对于100%的数据： $n \leq 500000, s < 2^{31}$, 所有权值 < 500