

算法实现题 3-21 有向树独立 k 中值问题

★问题描述:

给定一棵有向树 T ，树 T 中每个顶点 u 都有一个权 $w(u)$ ；树的每条边 (u, v) 也都有一个非负边长 $d(u, v)$ 。有向树 T 的每个顶点 u 可以看作客户，其服务需求量为 $w(u)$ 。每条边 (u, v) 的边长 $d(u, v)$ 可以看作运输费用。如果在顶点 u 处未设置服务机构，则将顶点 u 处的服务需求沿有向树的边 (u, v) 转移到顶点 v 处服务机构需付出的服务转移费用为 $w(u) * d(u, v)$ 。树根处已设置了服务机构，现在要在树 T 中增设 k 处独立服务机构，使得整棵树 T 的服务转移费用最小。服务机构的独立性是指任何 2 个服务机构之间都不存在有向路径。

★编程任务:

对于给定的有向树 T ，编程计算在树 T 中增设 k 处独立服务机构的最小服务转移费用。

★数据输入:

由文件 input.txt 给出输入数据。第 1 行有 2 个正整数 n 和 k 。 n 表示有向树 T 的边数； k 是要增设的服务机构数。有向树 T 的顶点编号为 $0, 1, \dots, n$ 。根结点编号为 0。接下来的 n 行中，每行有表示有向树 T 的一条有向边的 3 个整数。第 $i+1$ 行的 3 个整数 w_i, v_i, d_i 分别表示编号为 i 的顶点的权为 w_i ，相应的有向边为 (i, v_i) ，其边长为 d_i 。

★结果输出:

将计算的最小服务转移费用输出到文件 output.txt。

输入文件示例

```
input.txt
4 2
1 0 1
1 1 10
10 2 5
1 2 3
```

输出文件示例

```
output.txt
12
```