

支线剧情

【故事背景】

宅男 JYY 非常喜欢玩 RPG 游戏，比如仙剑，轩辕剑等等。不过 JYY 喜欢的并不是战斗场景，而是类似电视剧一般的充满恩怨情仇的剧情。这些游戏往往都有很多的支线剧情，现在 JYY 想花费最少的时间看完所有的支线剧情。

【问题描述】

JYY 现在所玩的 RPG 游戏中，一共有 N 个 **剧情点**，由 1 到 N 编号，第 i 个 **剧情点** 可以根据 JYY 的不同的选择，而经过不同的 **支线剧情**，前往 K_i 种不同的新的 **剧情点**。当然如果 K_i 为 0，则说明 i 号剧情点是游戏的一个结局了。

JYY 观看一个支线剧情需要一定的时间。

JYY 一开始处在 1 号剧情点，也就是游戏的开始。

显然任何一个剧情点都是从 1 号剧情点可达的。此外，随着游戏的进行，剧情是不可逆的。所以游戏保证从任意剧情点出发，都不能再回到这个剧情点。

由于 JYY 过度使用修改器，导致游戏的“存档”和“读档”功能损坏了，所以 JYY 要想回到之前的剧情点，唯一的方法就是退出当前游戏，并开始新的游戏，也就是回到 1 号剧情点。JYY 可以在任何时刻退出游戏并重新开始。

不断开始新的游戏重复观看已经看过的剧情是很痛苦，JYY 希望花费最少的时间，看完所有不同的支线剧情。

【输入格式】

从文件 *story.in* 中读入数据。

输入一行包含一个正整数 N 。

接下来 N 行，第 i 行为 i 号剧情点的信息；

第一个整数为 K_i ，接下来 K_i 个整数对， b_{ij} 和 t_{ij} ，表示从剧情点 i 可以前往剧情点 b_{ij} ，并且观看这段支线剧情需要花费 t_{ij} 的时间。

【输出格式】

输出到文件 *story.out* 中。

输出一行包含一个整数，表示 JYY 看完所有支线剧情所需要的最少时间。

【输入样例】

```
6
2 2 1 3 2
```

```
2 4 3 5 4
2 5 5 6 6
0
0
0
```

【输出样例】

```
24
```

【样例说明】

JYY 需要重新开始 3 次游戏，加上一开始的一次游戏，4 次游戏的进程是 $1 \rightarrow 2 \rightarrow 4$ ， $1 \rightarrow 2 \rightarrow 5$ ， $1 \rightarrow 3 \rightarrow 5$ 和 $1 \rightarrow 3 \rightarrow 6$ 。

【数据规模】

对于 30% 的数据满足 $N \leq 25, \sum_{i=1}^N K_i \leq 100$;

对于 70% 的数据满足 $N \leq 100, \sum_{i=1}^N K_i \leq 2000$;

对于 100% 的数据满足 $N \leq 300, 0 \leq K_i \leq 50, 1 \leq t_{ij} \leq 300, \sum_{i=1}^N K_i \leq 5000$ 。