

旅行售货员

(salesman.pas/c/cpp)

【问题描述】

某售货员小 T 要到若干城镇去推销商品, 由于该地区是交通不便的山区, 任意两个城镇之间都只有唯一的可能经过其它城镇的路线。小 T 可以准确地估计出在每个城镇停留的净收益。这些净收益可能是负数, 即推销商品的利润抵不上花费。由于交通不便, 小 T 经过每个城镇都需要停留, 在每个城镇的停留次数与在该地的净收益无关, 因为很多费用不是计次收取的, 而每个城镇对小 T 的商品需求也是相对固定的, 停留一次后就饱和了。每个城镇为了强化治安, 对外地人的最多停留次数有严格的规定。请你帮小 T 设计一个收益最大的巡回方案, 即从家乡出发, 在经过的每个城镇停留, 最后回到家乡的旅行方案。你的程序只需输出最大收益, 以及最优方案是否唯一。方案并不包括路线的细节, 方案相同的标准是选择经过并停留的城镇是否相同。因为取消巡回也是一种方案, 因此最大收益不会是负数。小 T 在家乡净收益是零, 因为在家乡是本地人, 家乡对小 T 当然没有停留次数的限制。

【输入格式】

输入的第一行是一个正整数 n , 表示城镇数目。城镇以 1 到 n 的数命名。小 T 的家乡命名为 1。第二行和第三行都包含以空格隔开的 $n - 1$ 个整数, 第二行的第 i 个数表示在城镇 $i + 1$ 停留的净收益。第三行的第 i 个数表示城镇 $i + 1$ 规定的最大停留次数。所有的最大停留次数都不小于 2。接下来的 $n - 1$ 行每行两个 1 到 n 的正整数 x, y , 之间以一个空格隔开, 表示 x, y 之间有一条不经过其它城镇的双向道路。输入数据保证所有城镇是连通的。

【输出格式】

输出有两行, 第一行包含一个自然数, 表示巡回旅行的最大收益。如果该方案唯一, 在第二行输出 “solution is unique”, 否则在第二行输出 “solution is not unique”。

【样例输入】

```
salesman.in
9
-3 -4 2 4 -2 3 4 6
4 4 2 2 2 2 2 2
1 2
```

1 3

1 4

2 5

2 6

3 7

4 8

4 9

【样例输出】

salesman.out

9

solution is unique

【样例说明】

最佳路线包括城镇 1, 2, 4, 5, 9。

【数据规模】

所有的净收益在正负 500 之间，所有的最大停留次数不大于 1000

30%的数据满足： $5 \leq n \leq 10$

60%的数据满足： $5 \leq n \leq 1000$

100%的数据满足： $5 \leq n \leq 100000$