第 29 届全国青少年信息学奥林匹克竞赛

上海市选拔赛

SHTSC 2012

第二试

竞赛时间：2012 年 4 月 29 日上午 7:30-12:00

题目名称	回家的路	排序	魔法树
英文名称	gohome	sorting	tree
源程序	gohome.pas 或 gohome.cpp	sorting.pas 或 sorting.cpp	tree.pas 或 tree.cpp
输入文件名	gohome.in	sorting.in	tree.in
输出文件名	gohome.out	sorting.out	tree.out
每个测试点时限	1 秒	1 秒	1 秒
内存限制	128M	128M	128M
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	否	是	否
题目类型	传统	传统	传统

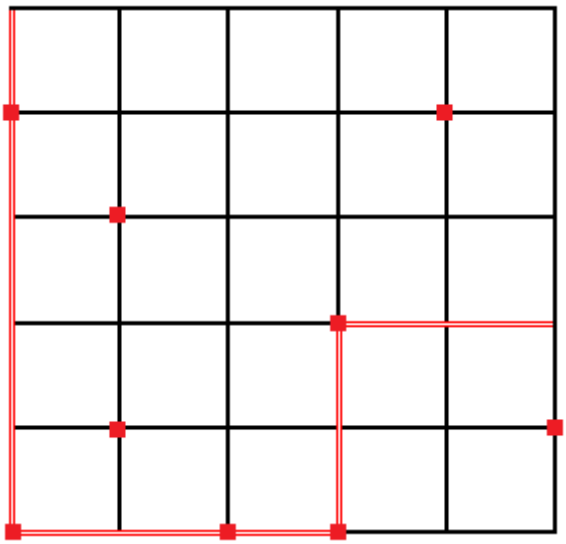
上海市科技艺术教育中心

回家的路

【问题描述】

2046 年 OI 城的城市轨道交通建设终于全部竣工，由于前期规划周密，建成后的轨道交通网络由 $2n$ 条地铁线路构成，组成了一个 n 纵 n 横的交通网。如下图所示，这 $2n$ 条线路每条线路都包含 n 个车站，而每个车站都在一组纵横线路的交汇处。

出于建设成本的考虑，并非每个车站都能够进行站内换乘，能够进行站内换乘的地铁站共有 m 个，在下图中，标上方块标记的车站为换乘车站。已知地铁运行 1 站需要 2 分钟，而站内换乘需要步行 1 分钟。Serenade 想要知道，在不中途出站的前提下，他从学校回家最快需要多少时间（等车时间忽略不计）。



【输入格式】

第一行有两个整数 n, m。

接下去 m 行每行两个整数 x, y，表示第 x 条横向线路与第 y 条纵向线路的交汇站是站内换乘站。

接下去一行是四个整数 x_1, y_1, x_2, y_2。表示 Serenade 从学校回家时，在第 x_1 条横向线路与第 y_1 条纵向线路的交汇站上车，在第 x_2 条横向线路与第 y_2 条纵向线路的交汇站下车。

【输出格式】

输出文件置有一行，即 Serenade 在合理选择线路的情况下，回家所需要的时间。如果 Serenade 无法在不出站换乘的情况下回家，请输出-1。

【输入样例 1】

```
2 1
1 2
1 1 2 2
```

【输出样例 1】

```
5
```

【输入样例 2】

```
6 9
2 1
2 5
3 2
4 4
5 2
5 6
6 1
6 3
6 4
1 1 4 6
```

【输出样例 2】

```
27
```

【输入样例 3】

```
6 10
2 1
2 5
3 2
4 4
5 2
5 6
6 1
6 3
6 4
6 6
1 1 4 6
```

【输出样例 3】

```
26
```

【数据规模】

对于 30%的数据，$n \leq 50, m \leq 1000$；
对于 60%的数据，$n \leq 500, m \leq 2000$；
对于 100%的数据，$n \leq 20000, m \leq 100000$；

排序

【问题描述】

众所周知，1~n 的全排列包含 n!个排列。通常情况下，我们在生成全排列时都按照他们的字典序生成的。而在本题中，我们就将要考虑一种特殊的全排列生成方式。

具体的，生成的全排列的顺序是由一个生成器决定的。

(1) 生成器本身也是一个 1~n 的排列：$a_1, a_2, \ldots, a_n$。

(2) 对于两个不相同的 1~n 的 $x_1, x_2, \ldots, x_n$、$y_1, y_2, \ldots, y_n$ 排列而言，首先找到最小的 i，使得 x_{ai} 与 y_{ai} 不相等。

(3) 根据(2)中选择的 i，如果 x_{ai} 在排列 $a_1, a_2, \ldots, a_n$ 中排在 y_{ai} 之前，那么 $x_1, x_2, \ldots, x_n$ 就会在 $y_1, y_2, \ldots, y_n$ 之前生成。

例如，当 n = 3，生成器为 132 时，1~n 的全排列的生成顺序为：123，132，321，312，231，213。

输入一个排列 $x_1, x_2, \ldots, x_n$，问，哪个生成器能使得这个排列在所有的排列中尽可能早的生成，哪个生成器能使得这个排列在所有的排列中尽可能晚的生成。如果有多种生成器能达到要求，那么请输出字典序最小的符合要求的生成器。

【输入格式】

输入的第一行是整数 n，第二行是 1~n 的一个排列 $x_1, x_2, \ldots, x_n$。

【输出格式】

输出的第一行是一个 1~n 的排列，表示让 $x_1, x_2, \ldots, x_n$ 尽早输出的生成器。
输出的第二行是一个 1~n 的排列，表示让 $x_1, x_2, \ldots, x_n$ 尽晚输出的生成器。
如果有多种生成器能达到要求，那么请输出字典序最小的符合要求的生成器。

【输入样例】

```
3
1 3 2
```

【输出样例】

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

【数据规模】

对于 30%的数据，有 $n \le 10$；
对于 50%的数据，有 $n \le 200$；
对于 90%的数据，有 $n \le 30000$；

对于 100%的数据，有 $n \leq 500000$。

【评分方法】

对于每个测试点：

- 仅第一行正确得 5 分；
- 仅第二行正确得 7 分；
- 全部正确得 10 分。

魔法树

【问题描述】

Harry Potter 新学了一种魔法：可以让改变树上的果子个数。满心欢喜的他找到了一个巨大的果树，来试验他的新法术。

这棵果树共有N个节点，其中节点0是根节点，每个节点u的父亲记为fa$[u]$，保证有fa$[u] < u$。初始时，这棵果树上的果子都被 Dumbledore 用魔法清除掉了，所以这个果树的每个节点上都没有果子（即0个果子）。

不幸的是，Harry 的法术学得不到位，只能对树上一段路径的节点上的果子个数统一增加一定的数量。也就是说，Harry 的魔法可以这样描述：

$$\text{Add}\ u\ v\ d$$

表示将点u和v之间的路径上的所有节点的果子个数都加上d。

接下来，为了方便检验 Harry 的魔法是否成功，你需要告诉他在释放魔法的过程中的一些有关果树的信息：

$$\text{Query}\ u$$

表示当前果树中，以点u为根的子树中，总共有多少个果子？

【输入格式】

第一行一个正整数N $(1 \le N \le 100000)$，表示果树的节点总数，节点以$0, 1, \dots, \text{N} - 1$标号，0一定代表根节点。

接下来$N - 1$行，每行两个整数a, b $(0 \le a < b < n)$，表示a是b的父亲。

接下来是一个正整数$Q(1 \le Q \le 100000)$，表示共有Q次操作。

后面跟着Q行，每行是以下两种中的一种：

1. A u v d，表示将u到v的路径上的所有节点的果子数加上d；$0 \le u, v < N, 0 < d < 100000$
2. Q u，表示询问以u为根的子树中的总果子数，注意是包括u本身的。$0 \le u < N$

【输出格式】

对于所有的Query操作，依次输出询问的答案，每行一个。答案可能会超过2^{32}，但不会超过10^{15}。

【输入样例】

```
4
0 1
1 2
2 3
4
A 1 3 1
Q 0
```

```
Q 1
Q 2
```

【输出样例】

```
3
3
2
```

【数据规模】

测试数据编号	特殊条件
1	果树呈链状，退化为一条直线
2	
3	
4	
5	
6	所有Add操作中的$u = 0$
7	
8	
9	无
10	