

训练赛 4

题目名称	串串	爬山	路径
源文件名	string.cpp/c/pas	climb.cpp/c/pas	path.cpp/c/pas
输入文件名	string.in	climb.in	path.in
输出文件名	string.out	climb.out	path.out
时间限制	1s	1s	1s
空间限制	64MB	128MB	64MB
优化开关	无	无	无

串串(string.cpp/c/pas)

题目描述

给一字符串 S , 和一个串 T . 一开始 T 是空串, 你每次可以从 S 的开头或末尾取出一个字符加到 T 的末尾. 要求最终的 T 的字典序尽可能小. 输出最后的串 T .

输入格式

- 第一行包含一个整数 N , 代表串 S 的长度.
- 接下来 N 行, 每行包含一个大写字母, 第 i 个字母就是 S 的第 i 个字母.
- S 只由大写字母构成.

输出格式

输出串 T , 每 80 个字符需要换一行.

样例

string.in	string.out
6 A C D B C B	ABCBCD

数据范围

- 存在 20% 的数据, $N \leq 10$.
- 存在 40% 的数据, $N \leq 100$.
- 存在 50% 的数据, $N \leq 400$.
- 对于 100% 的数据, $N \leq 2000$.

爬山(climb.cpp/c/pas)

题目描述

柠檬要爬山，每一段山路都可以抽象成一个线段，其两个端点分别为 $(x_1,y_1)/(x_2,y_2)$ ($x_1 < x_2, y_1 < y_2$)，但是 (x_2,y_2) 这个点是空的。题目保证任意两条线段都不交叉，相交(甚至在端点)，并保证第一段山路的起点 (x_1,y_1) 必然是 $(0,0)$ 。

现在柠檬要从原点开始爬，对于每一段山路都会一直爬到终点。到了终点 (x_2,y_2) 时，就会笔直地往下掉(即 y 不断减小, x 不变)，直到落到下一端山路上。若柠檬会掉到 $y=-\infty$ ，则爬山结束。现在 usbman 给出了所有山路的状况，柠檬想知道他能爬几段山路。

输入格式

第一行包含一个整数 N ，代表山路数目。

接下来 N 行，每行 4 个整数, x_1, y_1, x_2, y_2 ，描述了一段山路。

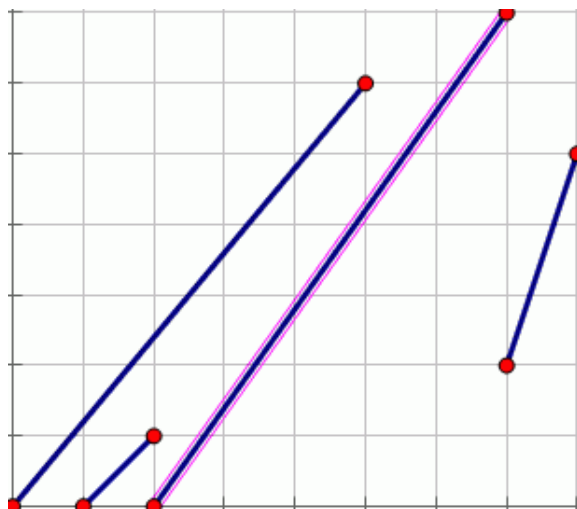
输出格式

输出一个整数代表答案。

样例

climb.in	climb.out
4 0 0 5 6 1 0 2 1 7 2 8 5 3 0 7 7	3

样例解释



数据范围

$0 \leq x1, y1, x2, y2 \leq 10^9$.

测试点	分值	N
1	8	≤ 10
2	8	≤ 10
3	8	≤ 50
4	8	≤ 200
5	10	≤ 1500
6	10	≤ 7000
7	8	≤ 30000
8	8	≤ 100000
9	8	≤ 100000
10	8	≤ 100000
11	8	≤ 100000
12	8	≤ 100000

路径(path.cpp/c/pas)

题目描述

有一棵 N 个点的树, 你需要用一些路径把所有边覆盖掉. 对于每条路径的每个端点, 它要么是一个叶子结点, 要么另外存在一条路径经过它, 且这个点不是那条路径的端点. 对于一条长度为 L 的路径, 它的花费为 L^2 . 求覆盖所有边的最小花费.

输入格式

第一行包含一个整数 N .

接下来 $N-1$ 行, 每行两个整数 u, v , 表示树中有一条连接 u, v 的边.

输出格式

输出一个整数代表答案.

样例

path.in	path.out
6 1 2 1 3 1 4 1 5 1 6	7

样例解释

有 4 条路径, 分别为 2-1-3, 1-4, 1-5, 1-6, 花费为 $2*2+1*1+1*1+1*1=7$.

数据范围

存在 25% 的数据, $N \leq 20$.

存在 45% 的数据, $N \leq 200$.

存在 65% 的数据, $N \leq 2000$.

对于 100% 的数据, $N \leq 50000$.