

省选模拟赛

2023-12-20

题目名称	整除	词典	蒲公英
源程序文件名	div	dictionary	dandelion
输入文件名	div.in	dictionary.in	dandelion.in
输出文件名	div.out	dictionary.out	dandelion.out
每个测试点时限	2.0s	2.0s	1.0s
内存限制	512MB	512MB	512MB
是否有子任务(subtask)	是	是	是
是否有Special Judge	否	否	是
题目类型	传统型	传统型	传统型

注意：

- 1. 编译选项 `g++ code.cpp -o code -lm -O2 -DONLINE_JUDGE -std=c++14`
- 2. 请不要开子文件夹，请尽量不要使用中文文件名（否则可能会被 OJ 忽略掉）

整除

Statement

求有多少个正整数 x 满足 $c_0 x^{a_0} + c_1 x^{a_1} + \dots + c_{n-1} x^{a_{n-1}}$ 能被 $x^0 + x^1 + \dots + x^{m-1}$ 整除, 其中 c 和 a 是两个给定的序列。

Task

input

第一行一个整数 $(1 \leq t \leq 10^5)$, 表示数据组数。

每组数据第一行两个整数 $n, m (1 \leq n \leq 10^5, 1 \leq m \leq 10^9)$, 接下来 n 行每行两个整数 c_i 和 $a_i (|c_i| = 1, 0 \leq a_i \leq 10^9)$ 。

所有数据中 n 的总和不超过 10^5 。

output

对于每组数据, 如果有无穷个正整数满足条件, 则输出 -1 , 否则输出满足条件的正整数个数。

Sample I

input

```
3
5 2
1 0
1 0
1 0
1 0
1 0
1 0
5 3
-1 2
-1 1
-1 0
1 1
-1 1
12 3
-1 0
-1 7
1 8
1 8
-1 4
-1 6
1 8
1 2
1 5
1 2
-1 9
1 5
```

output

```
1
-1
2
```

Constraints

Subtask 1 (10 points) : $t, n, m \leq 10$.

Subtask 2 (30 points) : $t \leq 10, n, m \leq 1000$.

Subtask 3 (60 points) : 无额外限制。

词典

Statement

一个 01 串 s 是**单词**当且仅当 s 中不包含两个连续的 0。

一个包含 n 个单词的**词典**是 n 个单词的集合，且满足其中任意一个单词都不是任意其他单词的前缀。

给定一个词典 D ，定义 01 串 s 的代价 $C(s) = \sum_{j=1}^k \lfloor (1 + \log_2 j) \rfloor$ ，其中 k 是 D 中满足 s 是 t 的前缀的单词 t 的数量。该词典 D 的代价即为所有 01 串的代价之和。

例如，考虑一个包含 4 个单词的词典 $\{0, 10, 110, 111\}$ 。这个词典的代价为

$$C(\epsilon) + C(0) + C(1) + C(10) + C(11) + C(110) + C(111) = 8 + 1 + 5 + 1 + 3 + 1 + 1 = 20$$

这里 ϵ 表示空串。

求包含 n 个单词的词典的代价的最小值。

Task

input

第一行一个整数 $(1 \leq t \leq 50000)$ ，表示数据组数。

接下来 t 行每行一个整数 $n (2 \leq n \leq 10^{15})$ ，表示词典包含的单词数量。

output

对于每组数据输出一个整数表示答案。

Sample I

input

```
6
2
4
10
5000
114514
1000000000000000
```

output

```
5
20
98
507842
21880717
1738413860843500846
```

Constraints

Subtask 1 (10 points) : $n \leq 5000$

Subtask 2 (20 points) : $n \leq 500\,000$

Subtask 3 (10 points) : $n \leq 30\,000\,000$

Subtask 4 (60 points) : 无额外限制。

蒲公英

Statement

给出一棵 n 个结点，直径上的边数不超过 4 的树，每个结点有一个 $1, 2, 3, \dots, n$ 中的唯一编号。给每条边编号为其连接的两个结点的编号的差的绝对值。

现需要对其结点重新编号，使边的编号不重复。判断无解，或给出一种重新编号的方案。

Task

input

第一行一个正整数 n ，表示给出的树的结点个数。

接下来 n 行，每行两个正整数 u_i, v_i ，表示一条树边连接的两个结点的编号。

output

如果不能构造方案，则在第一行输出字符串 No。

如果能够构造方案，则在第一行输出字符串 Yes，第二行输出 n 个正整数，第 i 个数表示原本编号为 i 的结点的新编号。

Sample I

input

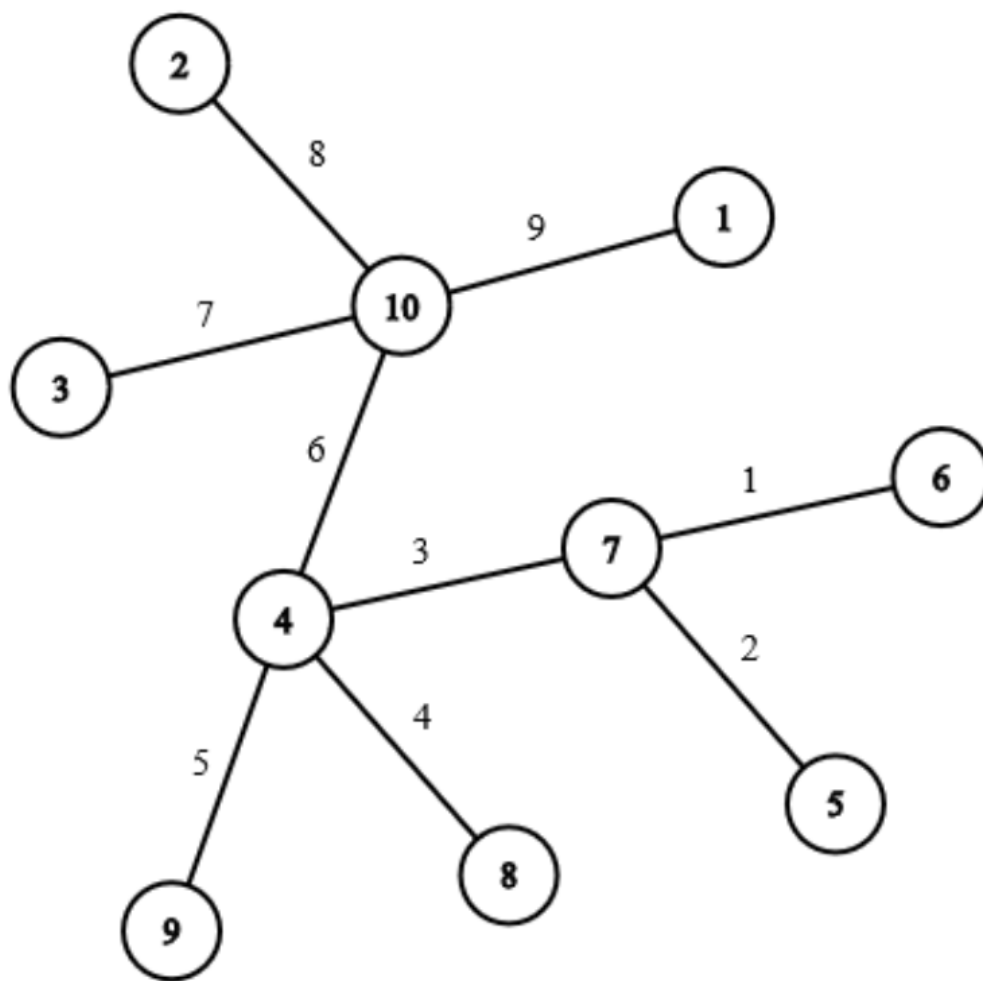
```
10
6 5
6 10
6 8
5 7
2 7
5 9
7 3
6 1
5 4
```

output

```
Yes
1 6 5 8 4 10 7 2 9 3
```

explanation

重新编号后树的形态如下图所示:



Constraints

对于所有测试点, 满足 $1 \leq n \leq 10^5$, $1 \leq u_i, v_i \leq n$, 保证给出的是一棵树且树直径上的边数不超过 4。

每个子任务的具体限制见下表:

子任务编号	子任务依赖	特殊限制	分值
1	无	$n \leq 10$	10
2	1	$n \leq 10^2$	20
3	1, 2	$n \leq 10^3$	20
4	无	树直径上边数不超过2	10
5	4	树直径上边数不超过3	30
6	1, 2, 3, 4, 5	无	10