



# Я Конкурс

## Алгоритмы и структуры данных. ПИиКТ. Осень 2025

⌚ 10 окт 2025, 18:12:16  
старт: 3 окт 2025, 20:49:29  
начало: 27 мар 2024, 12:49:22

### Н. Темпоральный катаклизм

Не решена

Ограничение времени	1 секунда
Ограничение памяти	50Mb
Ввод	стандартный ввод или input.txt
Выход	стандартный вывод или output.txt

Доктор Кто заметил, что каждое решение, принятое человечеством, создает новую временную ветвь, разделяя поток времени на бесчисленные альтернативные реальности. Эти ветви переплетаются в сложное древо, где каждая разветвка - это новый вариант будущего.

Однако недавний темпоральный катаклизм нарушил естественный ход времени, и теперь все эти ветви начинают нестабильно взаимодействовать друг с другом, угрожая привести к разрушению временного континуума. Чтобы предотвратить распад реальности, Доктору Кто необходимо перемещаться между ключевыми временными точками и восстанавливать порядок.

Но есть одна загвоздка: заряд ТАРДИС - машины времени Доктора - ограничен, и каждый прыжок по времененным линиям требует определённого количества энергии. Доктор должен оценить, хватит ли заряда для перемещения от одной точки к другой.

Помогите Доктору Кто справиться с последствиями темпорального катаклизма, путешествуя по ветвям времени и восстанавливая баланс в альтернативных реальностях!

#### Формат ввода

На вход подаётся следующее:

Первой строкой задаётся число  $N$  - количество временных точек (вершин) в древе, такое что  $1 \leq N \leq 10^5$

Следующие  $N-1$  строк описывают связи (ребра) между временными точками. Каждая строка содержит два числа  $i$  и  $v$ , которые обозначают соединение между временными точками  $i$  и  $v$ . Эти идентификаторы - натуральные числа от 0 до  $N-1$ , уникальные для каждой временной точки.

После этого задаётся число  $Q$  - количество запросов на перемещение, такое что  $Q \leq 10^6$  Следующие  $Q$  строк содержат описание запросов. Каждый запрос представлен тремя числами: два идентификатора временных точек  $i$  и  $v$ , между которыми Доктор хочет переместиться, и величина заряда  $T$ , доступного для этого перемещения, т.е. максимально возможное расстояние на которое можно переместиться, измеряемое в числе пройденных ребер, данная величина не превосходит  $N$ .

#### Формат вывода

На выходе для каждого запроса необходимо вывести одну строку:

"Yes", если Доктор может переместиться между двумя указанными временными точками при доступном заряде  $T$  (то есть если расстояние между точками меньше либо равно  $T$ ). "No", если заряд недостаточен для совершения перемещения (то есть если расстояние больше  $T$ ).

Каждый ответ выводится в отдельной строке, в том же порядке, в каком поступили запросы.

#### Пример 1

Ввод

```
7
0 1
1 2
1 4
4 5
4 6
2 3
3
6 5 3
3 5 2
0 6 1
```

Выход

```
Yes
No
No
```

#### Пример 2

[Ввод](#)[Выход](#)

Ввод	Выход
10	No
2 0	Yes
3 0	Yes
4 0	Yes
5 0	Yes
6 0	
7 0	
8 0	
9 0	
1 9	
5	
0 1 1	
1 2 3	
1 3 3	
1 5 3	
1 9 3	

## Ответ

Язык(Make) Clang 17.0.1 C++20

[Набрать здесь](#)    [Отправить файл](#)

```
1 #include <iostream>
2
3 int main()
4 {
5     /*
6     Пример ввода и вывода числа n, где -10^9 < n < 10^9:
7     int n;
8     std::cin >> n;
9     std::cout << n << std::endl;
10    */
11
12    return 0;
13 }
14
```

[Отправить](#)    Осталось 102 попытки< [Предыдущая](#) [Следующая](#) >

Посылок нет

[Соглашение](#) · [Документация](#) · [Поддержка](#)