

COMPARISON



СЕЗОН 2021/2022 – ШЕСТИ РУНД

Края на учебната година приближава и Хари има много работа за вършене. Вместо това той се опитва да превърти новата игра *String Comparison: Graph Edition*.

Правилата на играта са сравнително прости. Дадено ви е дърво, във всеки връх на което е записана буква от английската азбука. Трябва да можете да отговаряте на заявки от следния тип: сравнете лексикографски низа, който се получава от пътя между върховете **a** и **b** с низа, който се получава от пътя между върховете **c** и **d**. Низ от път между дадени върхове **v** и **u** се получава като долепим всички букви на пътя от **v** до **u** в реда, в който се срещат. Последния ъпдейт на играта добави и още един тип заявка: смени буквата във връх **v** на буква **c**.

Хари много иска да спечели играта, затова реши да си направи програма, която да отговаря на заявките вместо него. Предполагам знаете как завърши това негово начинание. Сега вие трябва да направите програма *comparison.cpp*, която да отговаря на заявките вместо него.

Вход

На първия ред от *comparison.in* са дадени числата **N** и **Q**, съответно броя върхове и заявки. На втория ред са дадени **N** символа, които са началните букви, записани във върховете, като **i**-тия от тях, номерирани от 1, отговаря за връх с номер **i**. На следващите **N – 1** реда са дадени ребрата на дървото. От последните **Q** се задават заявките във следния вид: **1, v, c** за заявка за промяна и **2, a, b, c, d** за заявка за сравнение.

Изход

За всяка заявка за сравнение трябва да изведете отговора на нов ред във файла *comparison.out*. Нека низът на пътя от **a** до **b** е **s₁**, а низът на пътя от **c** до **d** е **s₂**. То тогава, ако **s₁ < s₂** изведете 0, ако **s₁ > s₂** изведете 1, ако **s₁ = s₂** изведете 2.

Ограничения

$$1 \leq N, Q \leq 10^5$$

$$1 \leq a, b, c, d, v \leq N$$

Всички символи са букви от английската азбука.

Ограничение по време: 4.0 sec.

Ограничение по памет: 256 MB.

COMPARISON



СЕЗОН 2021/2022 – ШЕСТИ РУНД

Примерни тестове

Вход (comparison.in)	Изход (comparison.out)
6 6	0
adcbcb	1
1 6	2
3 6	0
2 4	
4 3	
5 3	
2 6 4 4 5	
2 1 5 1 2	
1 4 c	
2 6 4 6 5	
1 5 d	
2 5 1 2 3	