

Zadanie: BST

Własność BST

polish

ONTAK 2023, dzień 4. Dostępna pamięć: 256 MB. Limit czasu: 5 s.

03.07.2023

Dane jest *drzewo binarne*, czyli drzewo mające n wierzchołków ponumerowanych od 1 do n , w którym każdy wierzchołek może mieć co najwyżej dwoje dzieci (co najwyżej jedno lewe i co najwyżej jedno prawe dziecko). Oprócz numerów, wierzchołkom przyporządkowane są wartości – dla każdego $i = 1, 2, \dots, n$, wierzchołek i ma przyporządkowaną wartość a_i będącą liczbą całkowitą. Drzewo ma *własność BST*, jeśli wartość wierzchołka i jest większa od wszystkich wartości w jego lewym poddrzewie, a mniejsza od wszystkich wartości w jego prawym poddrzewie (czyli większa od lewego dziecka i wszystkich jego potomków, a mniejsza od prawego dziecka i wszystkich jego potomków).

Dane jest drzewo, które niekoniecznie ma własność BST. Możesz zmienić niektóre jego wartości na inne liczby całkowite – wyznacz najmniejszą liczbę zmian, po których drzewo będzie miało własność BST.

Wejście

W pierwszym wierszu wejścia podana jest liczba całkowita n ($1 \leq n \leq 300\,000$) – liczba wierzchołków drzewa. Drugi wiersz wejścia zawiera n liczb całkowitych a_1, a_2, \dots, a_n – wartości wierzchołków $1, 2, \dots, n$. Są to liczby całkowite, których wartość bezwzględna nie przekracza 10^8 . Kolejne n wierszy zawiera po dwie liczby całkowite x_i, y_i oznaczające numery wierzchołków będących odpowiednio lewym i prawym dzieckiem wierzchołka i . Jeśli odpowiednie dziecko nie istnieje, zamiast numeru podana jest liczba -1 .

Wyjście

Na wyjście wypisz dwa wiersze: pierwszy powinien zawierać minimalną liczbę k wartości, które trzeba zmienić w drzewie. Drugi wiersz powinien zawierać n liczb – taki ciąg nowych wartości wierzchołków, w którym dokładnie k wartości będzie się różniło od wejściowych, a powstałe drzewo będzie miało własność BST. Nowe wartości powinny być liczbami całkowitymi, których wartość bezwzględna nie przekracza 10^9 .

Jeśli istnieje więcej niż jedno rozwiązanie zmieniające dokładnie k wartości, można wypisać dowolne z nich.

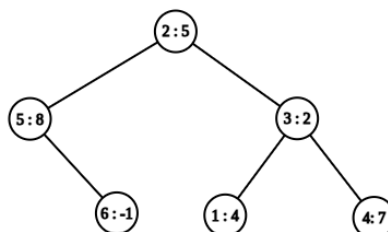
Dla danych wejściowych:

```
6
4 5 2 7 8 -1
-1 -1
5 3
1 4
-1 -1
-1 6
-1 -1
```

poprawnym wynikiem jest:

```
3
4 0 5 7 -7 -1
```

Wyjaśnienie przykładu:



Drzewo z przykładu (rysunek) nie ma własności BST – np. wartość w wierzchołku 3 równa jest 2, za to w wierzchołku 1, będącym lewym dzieckiem 3, jest większa wartość 4. Można przywrócić własność BST zmieniając wartości w wierzchołkach 2, 3 i 5 na odpowiednio 0, 5 i -7 ; istnieją też inne możliwe rozwiązania.

Ocenianie

Za każde podzadanie program otrzyma 50% przewidzianych za nie punktów, jeżeli we wszystkich testach poprawnie wypisze minimalną liczbę k zmian. Zawsze należy jednak wypisać nowy ciąg wartości (nie musi spełniać własności BST ani też różnić się na k miejscach, ale musi zawierać n liczb odpowiedniej wielkości).

Podzadanie	Ograniczenia	Punkty
1	$n \leq 20$	26
2	$n \leq 2000$	18
3	drzewo jest ścieżką (każdy wierzchołek ma tylko prawe dziecko)	24
4	bez dodatkowych ograniczeń	32

Завдання: BST

Własność BST

ukrainian

ONTAK 2023, день 4. Обмеження пам'яті: 256 MB. Ліміт часу: 5 s.

03.07.2023

Дане *бінарне дерево*, тобто дерево, яке має n вершин, пронумерованих від 1 до n , у якому кожна вершина може мати максимум двох дітей (максимум одну ліву і максимум одну праву дитину). Окрім номерів, вершинам присвоюються значення - для кожного $i = 1, 2, \dots, n$, вершині i присвоєне значення a_i , що є цілим числом.

Бінарне дерево задовольняє власність *BST* (*Binary Search Tree*), якщо для кожної вершини, її значення більше всіх значень вершин з її лівого піддерева, і менше всіх значень вершин з її правого піддерева.

Наше бінарне дерево прийшло з похибкою, і воно ще не обов'язково задовольняє власність BST. Але ми прагнемо до ідеалу, отже треба змінити якнайменше значень вершин дерева, щоб воно нарешті почало задовольняти властивість BST.

Вхідні дані

У першому рядку вводу подано ціле число n ($1 \leq n \leq 300\,000$) - кількість вершин дерева. Другий рядок вводу містить n цілих чисел a_1, a_2, \dots, a_n - значення вершин $1, 2, \dots, n$. Ці значення є цілими числами, абсолютне значення яких не перевищує 10^8 . Наступні n рядків містять по два цілих числа x_i, y_i , що означають номери вершин, які є відповідно лівою і правою дитиною вершини i . Якщо відповідна дитина не існує, замість номера подано число -1 .

Вихідні дані

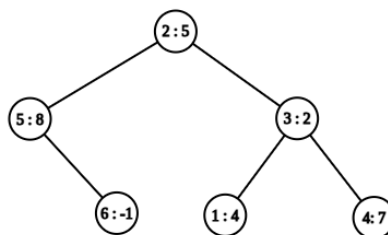
Розглянемо наступні вхідні дані:

```
6
4 5 2 7 8 -1
-1 -1
5 3
1 4
-1 -1
-1 6
-1 -1
```

Можливою коректною відповіддю може бути:

```
3
1 0 2 7 -7 -1
```

Пояснення прикладу:



Дерево з прикладу (малюнок) не має властивості BST - наприклад, значення у вершині 3 дорівнює 2, тоді як у вершині 1, що є лівою дитиною 3, є більше значення 4. Можна відновити властивість BST, змінивши значення в вершинах 2, 3 і 5 на відповідно 0, 5 і -7 ; також існують інші можливі рішення.

Оцінювання

За кожне підзавдання програма отримає 50% передбачених за нього балів, якщо в усіх тестах вона правильно виведе мінімальну кількість k змін. Проте, завжди потрібно вивести нову послідовність значень (вона не повинна задовольняти властивість BST, а також не обов'язково відрізнятися в k місцях, але повинна містити n чисел відповідного розміру).

Підзадача	Обмеження	Бали
1	$n \leq 20$	26
2	$n \leq 2000$	18
3	дерево є шляхом (кожна вершина має максимум праву дитину)	24
4	без додаткових обмежень	32