

# ВШЭ АиСД 2021. Сбалансированные деревья

4 дек 2021, 18:42:54

старт: 29 окт 2021, 12:00:00

финиш: 9 ноя 2021, 00:00:01

длительность: 10д. 12ч.

начало: 29 окт 2021, 12:00:00

конец: 9 ноя 2021, 00:00:01

## А. Проверка Красно-черного дерева (0.2)

Ограничение времени	1 секунда
Ограничение памяти	256.0 Мб
Ввод	стандартный ввод
Вывод	стандартный вывод

В данной задаче вам требуется проверить заданное бинарное дерево на красно-черность.

В дереве обычно присутствует фиктивная листовая вершина (NIL), которая по определению черная. Мы здесь будем придерживаться этого, но во входе она явна задана не будет (читайте описание входного формата).

Напомним свойства красно-черных деревьев:

1. Прежде всего, это дерево поиска, т.е. для любой вершины дерева  $node$ , любой вершины левого поддерева этой вершины  $left$  и любой вершины правого поддерева  $right$  выполняется строгое неравенство  $left.key < node.key < right.key$ .
2. Каждая вершина имеет красный или черный цвет.
3. Корень дерева и листья (NIL) имеют черный цвет.
4. Если вершина – красная, то ее родитель – черный.
5. Все простые пути из любой вершины  $x$  до листьев содержат одинаковое количество черных вершин.

Заметим, что черная вершина может иметь черного родителя.

### Формат ввода

В первой строке содержится целое число  $n$  ( $0 \leq n \leq 2 \cdot 10^7$ ) – размер дерева. Если  $n = 0$ , то на этом ввод закончен. Иначе на следующей строке содержится натуральное число  $root$  ( $1 \leq root \leq n$ ) – номер корневой вершины.

В следующих  $n$  строках содержатся описания вершин. Каждая строка описания имеет следующий формат:

number key left right color, где

1. number – номер соответствующей вершины ( $1 \leq number \leq n$ , номера не повторяются);
2. key – ключ вершины ( $-10^9 \leq key \leq 10^9$ );
3. left, right – номера левого и правого сына, соответственно (null, если соответствующий сын – это фиктивная вершина; иначе  $\in [1, n]$ );
4. color – цвет вершины (R или B).

Гарантируется, что заданный граф – бинарное дерево.

### Формат вывода

YES или NO в зависимости от корректности дерева.

### Система оценки

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
		$n$		
0	1	–	–	Тесты из условия.

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
1	5	$n \leq 10^5$	0	
2	2	$n \leq 2 \cdot 10^7$	0 – 1	Offline-проверка

Пример 1

Ввод

Вывод

5 1  
1 2 2 4 B  
2 0 null 3 R  
3 1 null null B  
4 3 null 5 R  
5 4 null null B

NO

Пример 2

Ввод

Вывод

5 1  
1 3 2 4 B  
2 1 null 3 R  
3 2 null null B  
4 4 null 5 B  
5 5 null null B

NO

Пример 3

Ввод

Вывод

3 1  
1 2 2 3 B  
2 3 null null R  
3 1 null null R

NO

Пример 4

Ввод

Вывод

7 7  
7 4 6 5 B  
6 2 4 3 R  
5 6 2 1 R  
4 1 null null B  
3 3 null null B  
2 5 null null B  
1 7 null null B

YES

Пример 5

Ввод

Вывод

1 1  
1 1 null null R

NO

```
1 #include <iostream>
2 #include <string>
3
4 struct TreeNode {
5     int value;
6     char color;
7     int numberLeftChildren;
8     int numberRightChildren;
9
10    explicit TreeNode(int new_value, char new_color, int number_left_children,
11                      int number_right_children) {
12        value = new_value;
13        color = new_color;
14        numberLeftChildren = number_left_children;
15        numberRightChildren = number_right_children;
16    }
17 };
18
19 class Tree {
20 private:
21     TreeNode *nodes_[2000001];
22     bool is_tree_correct_;
23     int size_;
24     int root_;
25
26     void deleteNode() {
27         for (int i = 0; i < size_; ++i) {
28             delete nodes_[i];
29         }
30     }
31
32     void insertNodes() {
33         int value;
34         int number;
35         std::cin >> number >> value;
36
37         std::string left_children;
```