

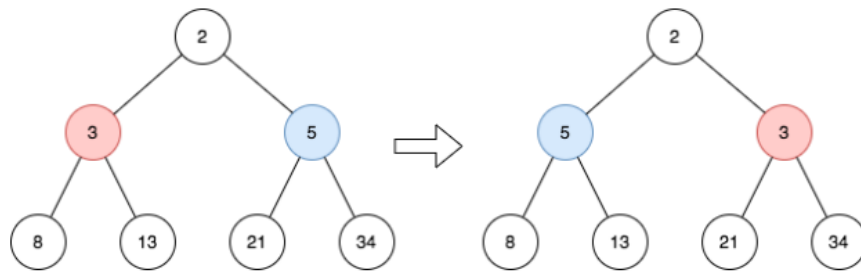
CS2023 - Aula de Ejercicios N° 6
Heider Sanchez
ACL: Juan Diego Castro
Semestre 2024-1

- Para este laboratorio ser formarán grupos de dos integrantes, pero se sugiere que cada estudiante intente primero resolver los ejercicios de forma **individual**.
- Queda totalmente prohibido el uso de foros de internet, ChatGTP, repositorios de GitHub, etc.
- Cualquier pequeño indicio de plagio será calificado con 0 (ya no habrá clemencia).
- Entregar los archivos **ejercicio1.cpp**, **ejercicio2.cpp** y **ejercicio3.cpp** sin comprimir. Cualquier otro archivo **no será revisado**.

Ejercicios

1. (7 pts) Dado el **root** de un árbol binario, invierta los nodos que se encuentran en cada uno de los **niveles impares** de la estructura. El **nivel** de un **nodo** se define como el número de aristas en el camino entre el **nodo** y el **root**.

- **Ejemplo 1:**

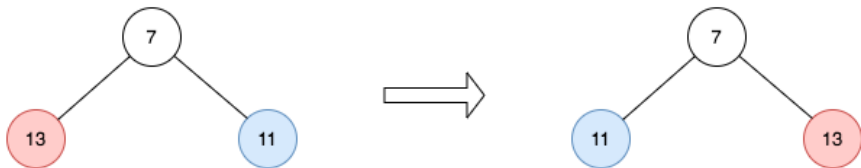


Input: root = [2,3,5,8,13,21,34]

Output: [2,5,3,8,13,21,34]

Explicación: El árbol solo tiene un nivel impar. Los nodos al nivel 1 (3 y 5) son intercambiados para convertirse en 5 y 3 (en ese orden).

- **Ejemplo 2:**



Input: root = [7,13,11]

Output: [7,11,13]

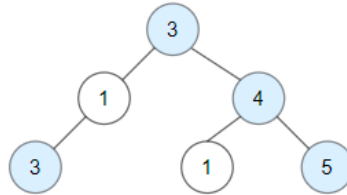
Restricciones:

- El **root** es un árbol perfecto (todos los padres tienen exactamente 2 hijos y todas las hojas están al mismo nivel).
- La cantidad de nodos en el árbol está entre $[1, 2^{14}]$

2. (7 pts) Dado el **root** de un árbol binario, se dice que un nodo \mathcal{X} es **bueno** si en el camino entre el **root** y \mathcal{X} no existen nodos con valores estrictamente mayores a \mathcal{X} .

Diseñe un algoritmo que cuente la cantidad de nodos **buenos** en **root**.

■ **Ejemplo 1:**



Input: root = [3,1,4,3,null,1,5]

Output: 4

Explicación: Los nodos celestes son **buenos**.

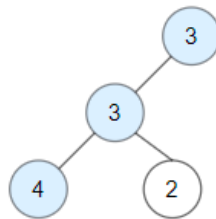
Root (3) Siempre es un nodo **bueno**.

Node 4 ->(3,4) es el valor máximo en el camino.

Node 5 ->(3,4,5) es el valor máximo en el camino.

Node 3 ->(3,1,3) es el valor máximo en el camino.

■ **Ejemplo 2:**



Input: root = [3,3,null,4,2]

Output: 3

Explicación:

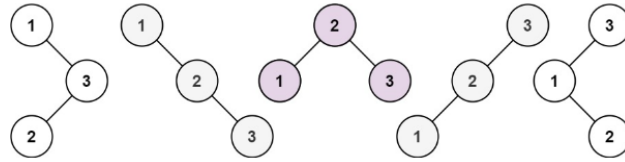
Node 2 ->(3, 3, 2) no es **bueno**, porque 3 es mayor que él.

Restricciones:

- La cantidad de nodos en el árbol está entre $[1, 10^5]$
- Cada nodo \mathcal{X} puede tomar valores entre $[-10^4, 10^4]$

3. (6 pts) Dado un entero n , retornar todos los BST's (árboles binarios de búsqueda) estructuralmente diferentes, los cuales tienen exactamente n nodos con valores no repetidos de 1 a n . Devuelva el resultado en cualquier orden.

■ **Ejemplo 1:**



Input: $n = 3$

Output: $[[1, \text{null}, 2, \text{null}, 3], [1, \text{null}, 3, 2], [2, 1, 3], [3, 1, \text{null}, \text{null}, 2], [3, 2, \text{null}, 1]]$

Restricciones:

- $1 \leq n \leq 8$