

Taller- Árboles AVL - Parejas

Juan Niño-2240040

Francesco Hernandez-2240032

1. URL Repositorio: https://github.com/juanmanino2240040/Est-Datos_taller_arboles_2_AVL

2. **Nuestra Implementacion:**

Se agregaron los metodos delete(self, value), _delete_recursive(self, node, value), _inorden(self), _inorden_recursivo(self, node, resultado), _mostrar_arbol(self), y _mostrar_arbol_recursivo(self, node, indent, last).

Nuestra implementacion consiste en corregir el error de asignacion en el metodo insert recursivo y agregar las funcionalidades que pedía el problema, y no tenia.

Estas eran:

- Los metodos de eliminacion: hicimos dos metodos, uno que indica donde empieza la busqueda del nodo a eliminar y llama al metodo recursivo, y otro el metodo de eliminacion aplicando la recursividad
- Los metodos de recorrido Inorden: de igual forma, tiene el metodo que crea la lista y llama la metodo recursivo, indicando en que nodo empieza, la raiz, e imprimiendo el resultado

3. **Errores en el código:**

Solo había un error, pero era fatal. En el metodo insert recursivo, al compara los balances luego de insertar un nuevo nodo, estaba el problema. El error era que dentro de cada if, cuando hacia un rotate para left o para right, llamaba a las funciones de rotación sin poner el statement RETURN.

Las funciones rotate() reciben un nodo (A) que tiene un hijo(B), y este a su vez, un hijo (C) (cambia si es hijo derecho o izquierda si es rotate right o left), y retornan el nodo que ahora está en la posicion del nodo inicial, en nodo B.

En la función insert recursiva, al no tener return, se realizaban las rotaciones y no permitia devolver el nodo que ahora es raiz de ese sub-árbol rotado, causando que en la recursividad se perdieran los nodos, ya que en los ciclos, no se asignaban los nodos rotados a una posicion, y quedaban perdidos. Cuantos más nodos habia y mas ciclos recursivos se realizan, más nodos se extraviaban

4. **Casos de Prueba:**

Se realizaron las comprobaciones con la lista proveída por el codigo, [10, 20, 30, 40, 50, 25]. Se hizo la comparación lado a lado con el simulador de la Universidad de San Francisco para estar seguros de los resultados.

Usando un for each se agregaron los datos de la lista en el arbol y usamos el metodo de mostrar arbol para comprobar que los nodos estén en su correcto lugar y sigan las propiedades AVL y BB.

Luego, probamos eliminando la raiz para ver como cambiaba el arbol, y efectivamente logró ordenarlo correctamente