

OBSERVACIONES DEL LA PRACTICA

Estudiante 1 Sergio Villamarin 202123954

Estudiante 2 Juanita Ramirez 202121634

Preguntas de análisis

1) ¿Qué diferencia existe entre las alturas de los dos árboles (BST y RBT)?

La altura del BST puede llegar a ser igual que el número de elementos, ya que este no se encuentra balanceado. Mientras que un RBT siempre está balanceado, lo que significa que la altura del RBT es siempre $\log(n)$. Por lo que generalmente los RBT presentan menor altura que los BST, aunque existe la posibilidad de que si no es el peor de los casos y este balanceado, sean iguales.

2) ¿Percibe alguna diferencia entre la ejecución de los dos árboles (RBT y BST)? ¿Por qué pasa esto?

Los BST pueden degradarse a un árbol lineal en el peor de los casos, lo que resulta en un tiempo de ejecución de $O(n)$ para las operaciones de búsqueda, inserción y eliminación. Por otro lado, los RBT están siempre balanceados, lo que garantiza un tiempo de ejecución de $O(\log n)$ para estas mismas operaciones. Entonces es posible evidenciar que los RBT presentan menor tiempo de ejecución.

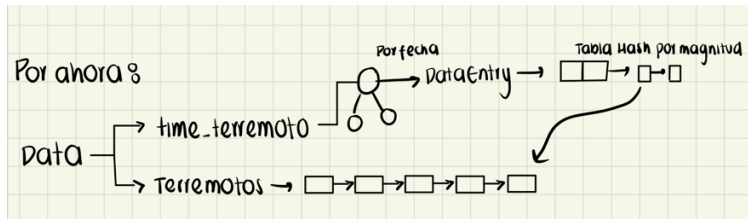
3) ¿Existe alguna diferencia de complejidad entre los dos árboles (RBT y BST)? Justifique su respuesta.

En términos de complejidad, también existe una diferencia entre los BST y los RBT. Los BST tienen una complejidad de tiempo de $O(n)$, en el peor de los casos, mientras que los RBT mantienen una complejidad de tiempo de $O(\log n)$ en todos los casos. Por lo que, es posible decir que los RBT presentan una complejidad mas optima, ya que en el peor de los casos tambien presentan complejidad de $O(\log n)$. Mientras que con el RBT varia dependiendo si es el peor de los casos o no

4) ¿Existe alguna manera de cargar los datos en un árbol RBT de tal forma que su funcionamiento mejore? Si es así, mencione cuál.

Si se cargan de forma ordenada, es probable que mejore la complejidad y por ende, mejore el funcionamiento.

Diagrama Avance Reto 3:



Se presentan listas, mapas, tabla de hash y listas enlazadas. Se utilizan mapas ya que estos son óptimos para buscar entre dos datos, tablas de hash ya que agilizan buscar algún dato y las listas en general porque permiten guardar o almacenar los datos de forma que se acceden a estos fácilmente .