

Estructuras de datos y algoritmos avanzados

Laboratorio 2

Francisco Flores

1. Introducción

En el presente informe se muestran la implementación y análisis (teórico y práctico) de dos estructuras de datos basados en el ADT Tree. En particular, un análisis de las operaciones buscar e insertar para BST y Splay Tree.

2. Estructuras de datos y algoritmos

Se implementa un árbol binario de búsqueda (BST) con las operaciones de insertar y buscar, y un Splay-Tree que hereda del BST anterior y soporta las mismas operaciones, además de realizar Splay después de cada búsqueda.

Las operaciones se describen a continuación:

- a. insertar: inserta un nuevo elemento en el árbol.
- b. buscar: busca un elemento en el árbol.
- c. splay: Lleva al nodo buscado a la raíz del árbol.

3. Implementación

Se han implementado estas estructuras de datos y algoritmos usando el lenguaje de programación JAVA. Splay-Tree hereda de BST Los algoritmos para insertar, buscar y splay se describen como sigue:

1. `insertar(k)`: agrega k a la raíz del árbol si esta es nula, sino, lo agrega en el hijo derecho en caso de k ser mayor al elemento de la raíz o en el hijo izquierdo en caso contrario (si es igual no lo agrega). Y se repite la misma operación hasta insertar el elemento. Costo $O(\log(n))$ en promedio y $O(n)$ en peor caso.
2. `buscar(k)`: busca el elemento k en el árbol y retorna verdadero si lo encuentra y falso si no. Tiene la misma complejidad que la operación insertar. Salvo en el Splay-Tree que al preguntar consecutivamente por un elemento se vuelve constante gracias a la operación de splay.
3. `splay()`: hace rotaciones de Zig-Zig y Zig-Zag para llevar el valor consultado k a la raíz del árbol. Costo $O(\log(n))$.

4. Resultados