

Estructura de Datos para ubicar directorios y archivos en un almacenamiento

***Laura Sánchez Córdoba
Luis Carlos Rodríguez Zúñiga
Manuela Valencia Toro***

Medellín, 30 de Octubre de 2017

Estructuras de Datos Diseñada

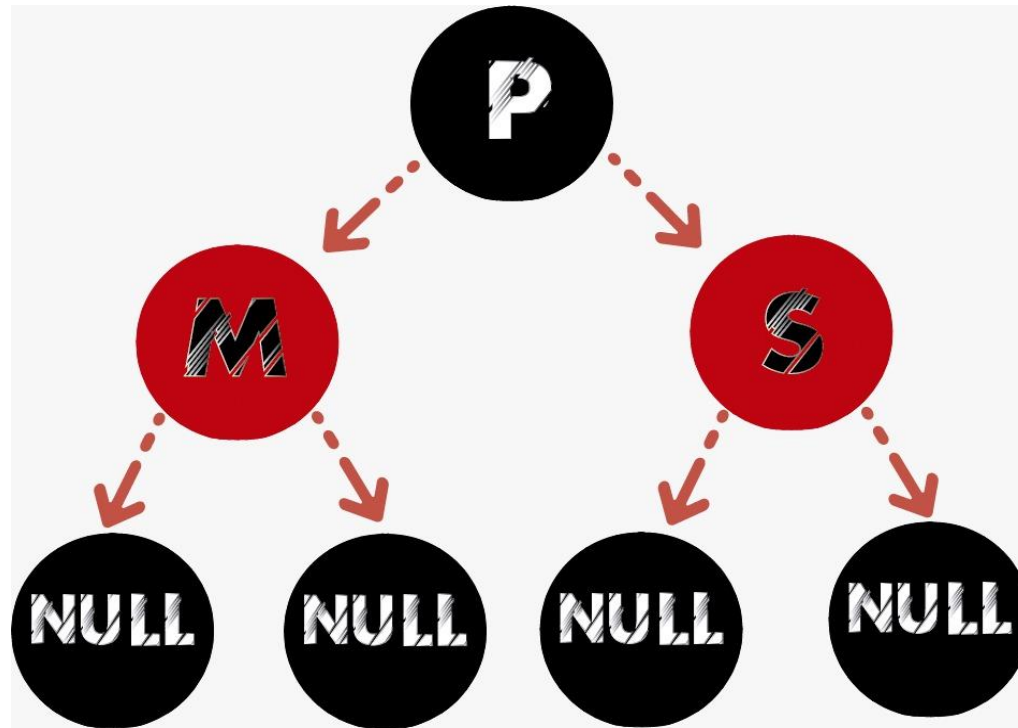


Gráfico 1: Arbol Rojo Negro. Ejemplo de la organización de los datos tipo String, de esta manera se organizan los archivos basados en el nombre, cada nodo tiene el nombre de la carpeta como llave (son los que se organizan alfabéticamente) y como valor el dato que el usuario quiera buscar (tamaño, dueño, ubicación).

Estructuras de Datos Diseñada

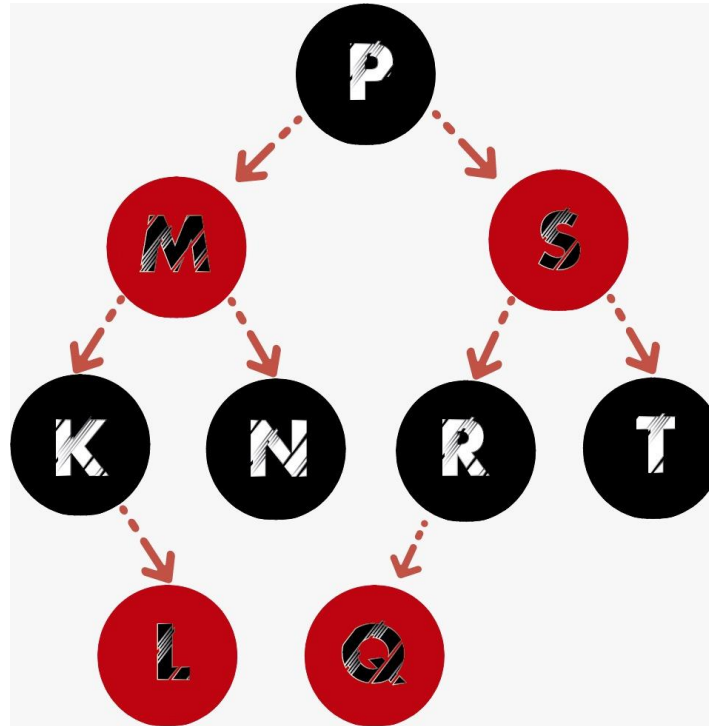


Gráfico 2: Así se van almacenando los datos y se balancea el árbol según las propiedades del árbol, cada nodo puede ser archivo o carpeta.

Operaciones de la Estructura de Datos

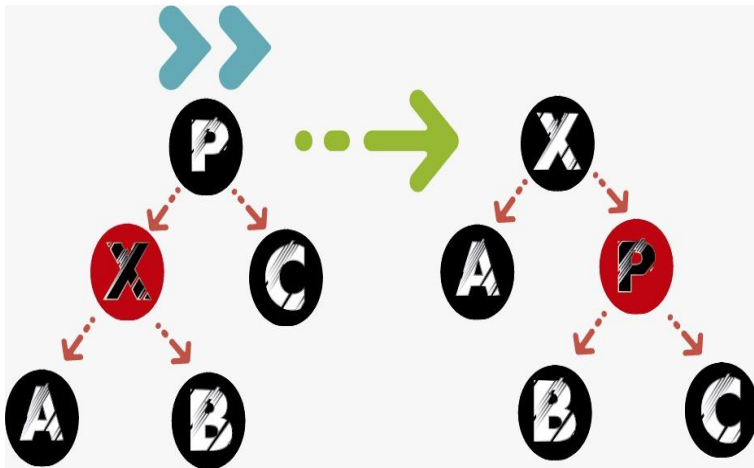


Gráfico 3: Rotación hacia la derecha

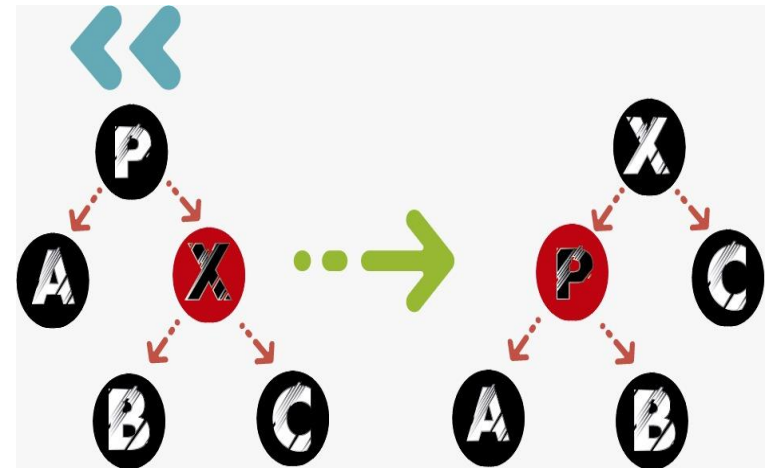


Gráfico 4: Rotación hacia la izquierda

Operaciones de la Estructura de Datos

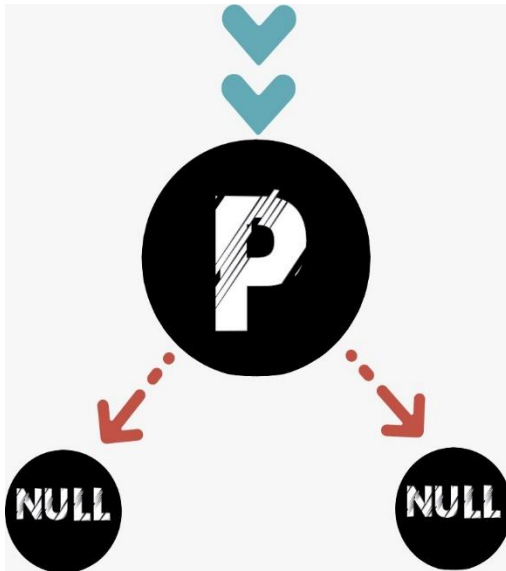


Gráfico 5: Primer caso de inserción, cuando el nodo ingresa a la raíz

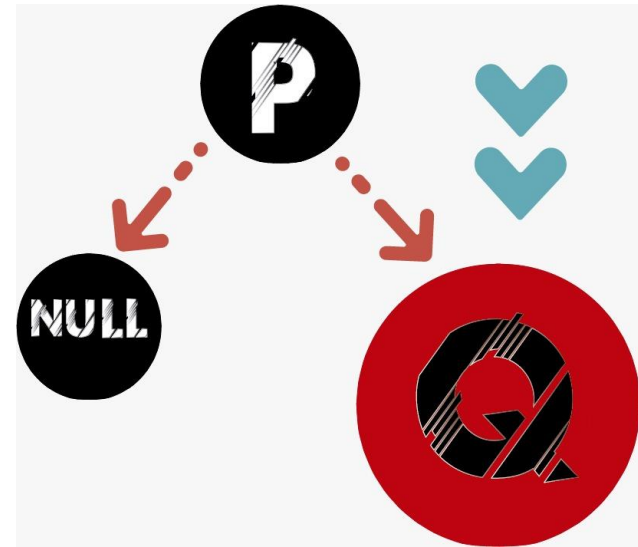


Gráfico 6: Segundo caso, cuando el nodo a insertar es hijo de la raíz

Operaciones de la Estructura de Datos

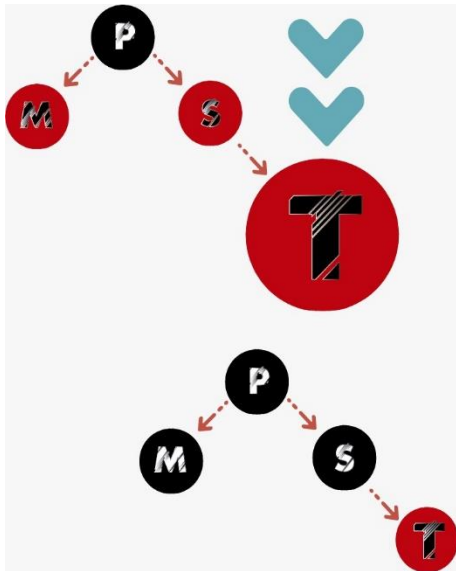


Gráfico 7: Tercer caso, cuando modifica los colores para que las 4 propiedades se cumplan.

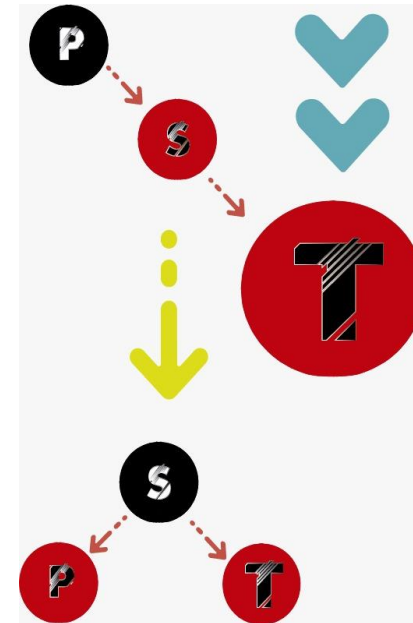


Gráfico 8: Rotación para balanceo de árbol

Operaciones de la Estructura de Datos

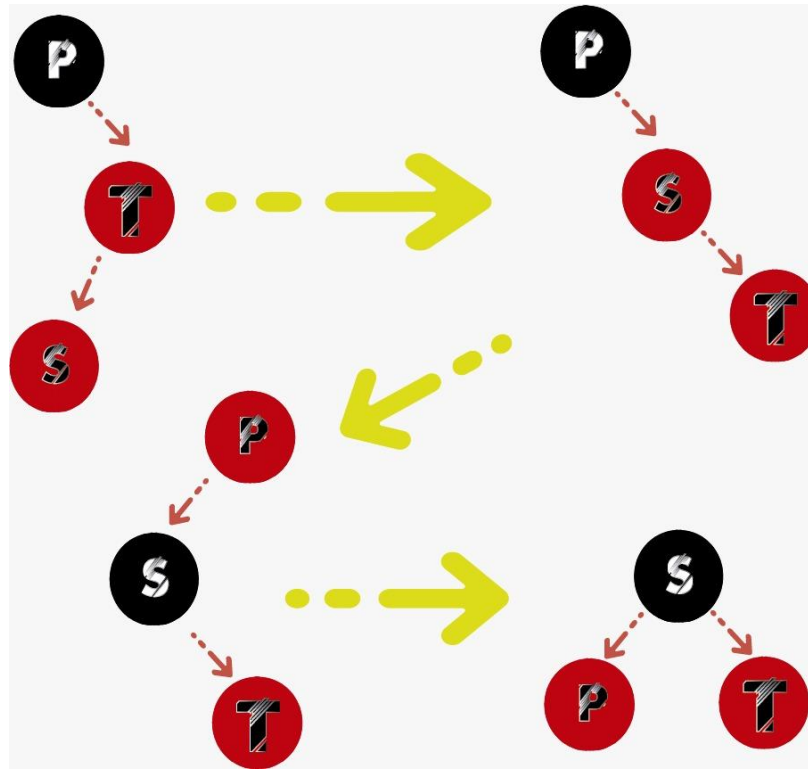


Gráfico 9: Quinto caso, una doble rotación para poder equilibrar el arbol, y cumplir las 4 propiedades.

Operaciones de la Estructura de Datos

	Conjunto de Datos 1	Conjunto de Datos 2
Consumo de Memoria	16 mb	36 mb

Tabla 1: Consumo de Memoria.

Método	Complejidad
Lectura y almacenamiento	$O(n \log n)$
Búsqueda por nombres	$O(n \log n)$
Búsqueda por tamaño	$O(\log n)$
Búsqueda por dueño	$O(\log n)$

Tabla 2: Complejidad de los metodos

Operaciones de la Estructura de Datos

Métodos	Mayor tiempo	Menor tiempo	Tiempo promedio
Lectura y almacenamiento	147 ms	133 ms	140 ms
Búsqueda por nombres	592559 ns	117115 ms	354837 ns
Búsqueda por tamaño	76022 ns	68625 ns	72323 ns

Tabla 3: Tiempos promedio del algoritmo al ejecutar los diferentes métodos de la estructura.

Criterios de Diseño de la Estructura de Datos

- En la solución del problema se utilizó la implementación por defecto de java del árbol rojo-negro. (TreeMap)
- La operación de inserción en el árbol tiene una complejidad de: $n \log n$
- La implementación del árbol permite que el tiempo de búsqueda sea más óptimo
- La programación en Java permitió que el almacenamiento de los datos no requiriera muchos recursos, sobre todo en cuestión de memoria.

Consumo de Tiempo y Memoria

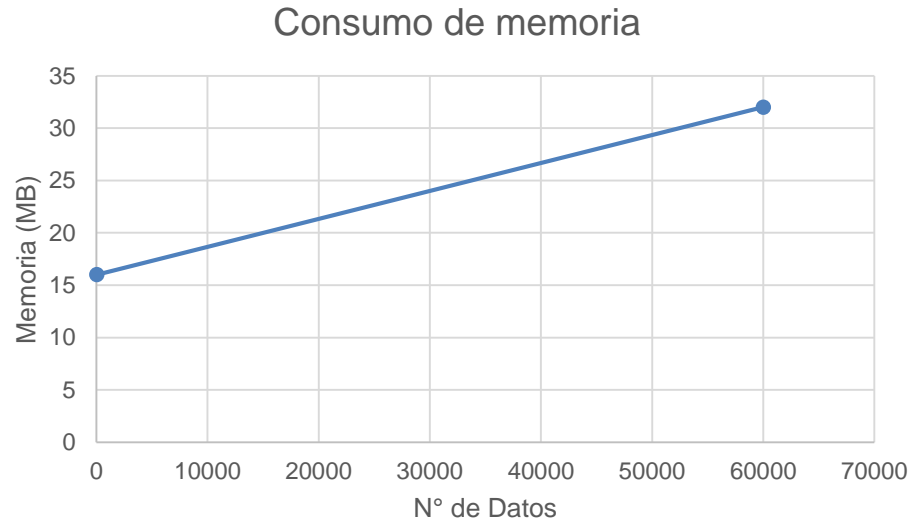


Gráfico 10: Consumo de memoria durante la ejecución del programa con distintos conjuntos de datos

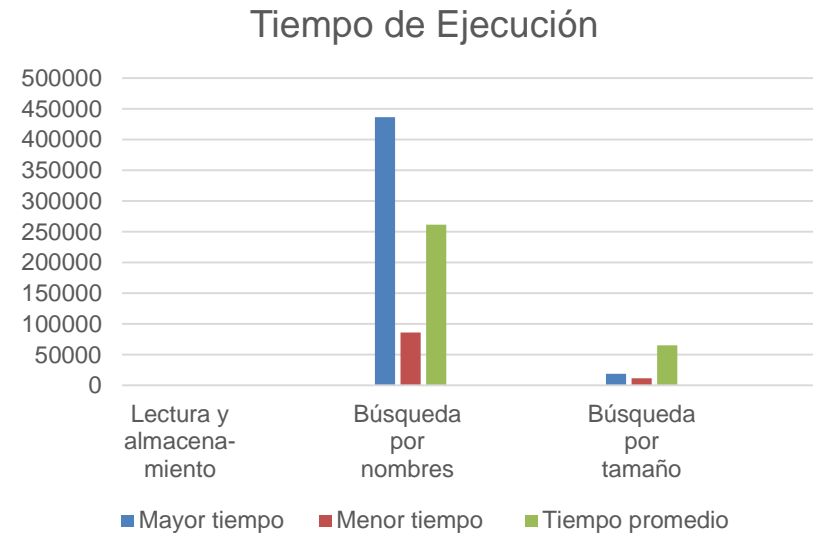


Gráfico 11: Comparación de tiempos de ejecución

Software Desarrollado

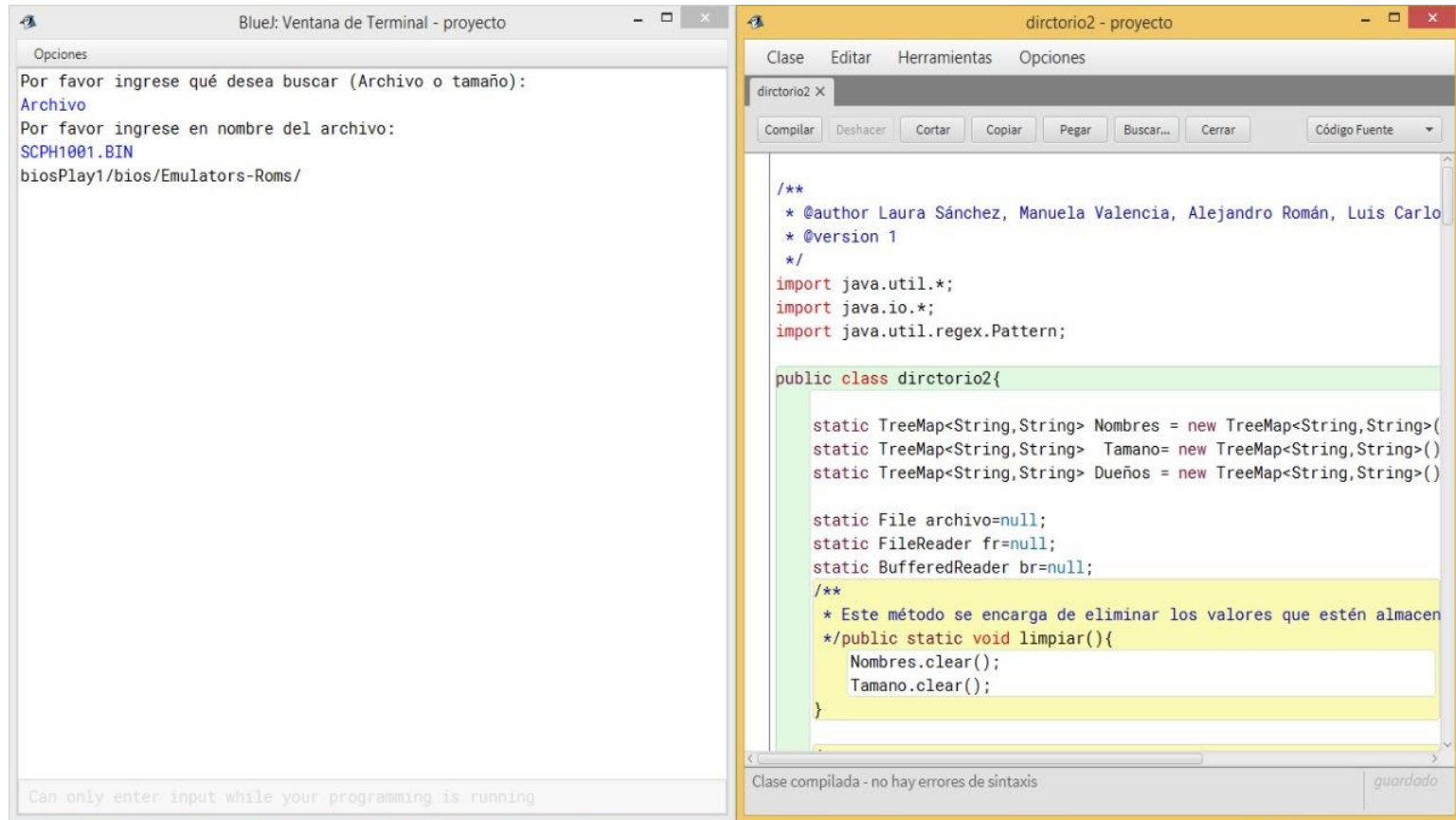


Gráfico : Sistema en funcionamiento