

Árbol Binario de Búsqueda

Se pide implementar una Árbol Binario de Búsqueda (ABB) en el lenguaje de programación C. Para ello se brindan las firmas de las funciones públicas a implementar y **se deja a criterio del alumno la creación de las funciones privadas del TDA** para el correcto funcionamiento del **ABB** cumpliendo con las buenas prácticas de programación.

El TDA entregado deberá compilar y **pasar las pruebas dispuestas por la cátedra sin errores**, adicionalmente estas pruebas deberán ser ejecutadas sin pérdida de memoria.

Aclaraciones de la implementación

Esta implementación de **ABB** incluye una función de comparación que permite insertar cualquier tipo de puntero en el mismo. **El ABB no tiene idea de qué es lo que almacena el usuario**, simplemente es un contenedor de datos que sigue reglas definidas. Mediante el comparador el **ABB** es capaz de establecer la relación de orden de los diferentes punteros, sin necesidad de tener información extra. **Recuerde que los comparadores devuelven 0, >0 o <0** según la relación de los elementos comparados (no necesariamente 0, 1, -1).

Por último, tenga en cuenta que las pruebas de de la cátedra suponen que la implementación acomoda **los elementos menores del ABB del lado izquierdo y los mayores del lado derecho**, y que al borrar nodos con dos hijos no nulos, **se reemplaza dicho nodo con el predecesor inorden**.

NO está permitido modificar los archivos **.h**.

A incluir en el informe

- Explique teóricamente (y utilizando gráficos) qué es una árbol, árbol binario y árbol binario de búsqueda. Explique cómo funcionan y de ejemplos de utilización de cada uno.
- Explique la implementación de ABB realizada y las decisiones de diseño tomadas (por ejemplo, si tal o cuál funciones fue planteada de forma recursiva, iterativa o mixta y por qué, que dificultades encontró al manejar los nodos y punteros, reservar y liberar memoria, etc).
- Explique la complejidad de las operaciones implementadas para el TDA.