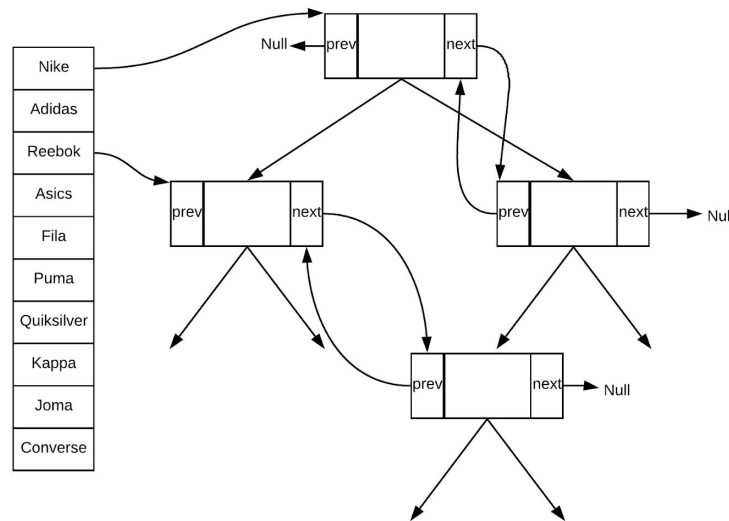


Explicación

La estructura de datos que usamos para gestionar el problema de esta práctica utiliza un Árbol AVL en el que se guardan, ordenados por su **id**, todos los productos del stock y una lista doblemente enlazada para cada marca (Nike, Adidas, Reebok, Asics...) que contiene punteros a los productos que son de esa marca en concreto. El tipo **estoc** contiene un Árbol AVL de punteros a productos y un array indexado por las marcas, que están guardadas en un tipo enumerado llamado **marca**, de listas doblemente enlazadas de punteros a productos.



En la imagen, para simplificar, los nodos del árbol ya son el mismo producto y no un tipo nodo del package del Árbol AVL que contiene un elemento “item” que en este caso sería un puntero a un producto. El tipo producto ya incluye dos punteros a producto, uno que se llama “prev” y otro “next”, además de toda su información propia. El Árbol AVL está implementado con punteros que permiten el uso de la memoria dinámica. El paquete de árbol AVL incluye un iterador que se usa para hacer el recorrido en inorden, que está implementado usando una pila. La pila está implementada con punteros. Las listas doblemente enlazadas también usan punteros.

Justificación

Usamos un árbol como parte principal de la estructura debido a que tiene un coste de consulta, inserción y de eliminación (media) de $O(\log n)$ y hay dos funciones que son la inserción de un producto y la eliminación. Además, es posible utilizar un árbol binario de búsqueda porque los productos tienen claves numéricas únicas y que se utilizan para guardarlos de forma ordenada, cosa que pide el procedimiento de imprimir todos los productos. El procedimiento de imprimir todos los productos de una marca implica que tengamos que recorrer una lista de los productos que son de la marca, ya que si hiciéramos un recorrido del árbol entero comprobando caso por caso si pertenecen a la misma el coste computacional sería demasiado alto. Por eso, cada marca tiene una lista doblemente enlazada (para simplificar las eliminaciones) de punteros a productos. Se usan punteros en vez de productos para no duplicar información.