

**Pregunta única:**

**Con los códigos Java implementados en clase, haga lo siguiente:**

**Vamos a comparar el rendimiento de una lista contra el rendimiento de un árbol binario de búsqueda. Cree una lista simplemente enlazada y un árbol binario de búsqueda, para almacenar números enteros. Cree 10 mil números enteros aleatorios entre 1 y 100, e insértelos tanto en la lista como en el árbol.**

**Ahora vamos a medir el tiempo de buscar un valor aleatorio en ambas estructuras. Para medir el tiempo de ejecución en Java, se recomienda usar `System.nanoTime()`, tal como se indica a continuación:**

```
long t0 = System.nanoTime();  
// Aquí ponga el método para buscar en la lista  
long t1 = System.nanoTime();  
// tiempo transcurrido en milisegundos:  
long tiempoBuscaLista = (t1 - t0)/1000000;  
  
long t2 = System.nanoTime();  
// Aquí ponga el método para buscar en el árbol  
long t3 = System.nanoTime();  
// tiempo transcurrido en milisegundos:  
long tiempoBuscaArbol = (t3 - t2)/1000000;
```

**Repita 10 veces la medición, indicando el tiempo promedio de una búsqueda de 10 números aleatorios en cada estructura.**

**Notas:**

- No es recomendable almacenar los números en un arreglo, sino generarlos en una variable e insertarlos directamente en ambas estructuras.
- Puede definir en cada clase de estructura (Lista, árbol) solo los métodos que vaya a usar.
- Salve su programa antes de ejecutarlo. Si tarda mucho, o parece "trancarse", pruebe con menor cantidad de números.
- El trabajo puede realizarse en equipos hasta de 4 personas.
- Solo se acepta una entrega por equipo. Recuerde indicar los integrantes.