

### Hoja de trabajo No. 7

**Realizar:** Un diccionario Inglés - Español.

**Realizarse:** en tríos.

**Objetivos:**

- Implementación de árboles binarios de búsqueda (BST).
- Uso de los BST para un map basado en árboles.

**Definiciones importantes:**

**Binary Tree (BT):** Un árbol binario (en inglés binary tree) es un árbol ordenado de, a lo sumo, grado 2.

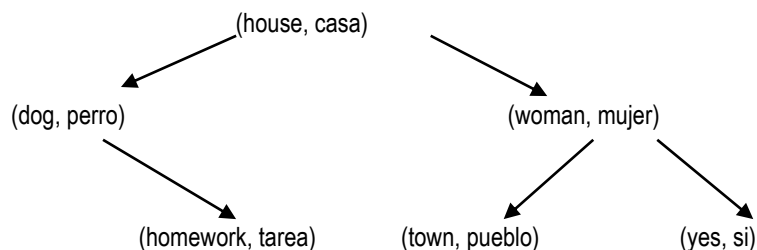
**Binary Search Tree (BST):** Un árbol binario es un árbol binario de búsqueda si es trivial, o si cada nodo es simultáneamente mayor o igual que cada valor en su subárbol izquierdo y es menor o igual que cada valor en su subárbol derecho.

**Programa a realizar:**

Su programa implementará un diccionario Inglés – Español. Para alimentar el diccionario recibirá el archivo diccionario.txt que contiene ASOCIACIONES con la palabra en inglés y su equivalente en español. Ejemplo del contenido de este archivo es:

(house, casa)  
(dog, perro)  
(homework, tarea)  
(woman, mujer)  
(town, pueblo)  
(yes, si)

Usted debe formar un árbol binario de búsqueda (BST) con estas asociaciones, basado en la palabra en inglés. Por ejemplo el árbol formado con las asociaciones mostradas es:



Al finalizar de procesar todas las asociaciones del archivo, recorra el árbol donde se guardó el diccionario en In-order. Debe producir la colección de palabras ordenadas por inglés. Con los datos de este ejemplo: (dog, perro) (homework, tarea) (house, casa) (town, pueblo) (woman, mujer) (yes, si)

Además su programa debe procesar el archivo texto.txt que contiene un documento en inglés. Usted debe traducir cada palabra al español. Si la palabra no está en su diccionario, entonces escribe la palabra original en inglés, encerrada entre asteriscos.

Por ejemplo si el texto.txt que recibe es:

The woman asked me to do my homework about my town.

Produce la salida en la terminal:

\*The\* mujer \*asked\* \*me\* \*to\* \*do\* \*my\* tarea \*about\* \*my\* pueblo.

NOTA: no debe importar si la palabra esta en mayúscula o minúscula para poder encontrarla en el diccionario.

**Tareas:**



- Implemente la clase BinaryTree <E> para hacer las operaciones de un árbol binario. Puede usar como referencia la mostrada en el libro o en algún sitio Internet. Recuerde siempre poner el lugar de donde tomó la idea.<sup>1</sup>
- Implemente la clase Association <K, V> que representa las asociaciones del diccionario.
- Haga pruebas unitarias para el método que inserta un elemento en el árbol y para el método que busca un elemento en el árbol.

Debe subir a Blackboard todos los productos elaborados y los enlaces a su repositorio de github (o equivalente).

#### Calificación:

Aspecto	Puntos
Uso del repositorio: existen más de tres versiones guardadas, la última versión es igual a la colocada en el Blackboard.	10
Implementación de clase Association y BinaryTree	10
Recorrido in-order del árbol: mostrar todas las relaciones ordenadas por la palabra en inglés.	30
Traducción del archivo de entrada.	40
Pruebas unitarias de los métodos para insertar y buscar una asociación en el árbol.	10
<b>TOTAL:</b>	<b>100</b>

#### Referencias:

Interfaz Comparable: <https://docs.oracle.com/javase/8/docs/api/java/lang/Comparable.html>

Bailey, Duane A. (2007). Java Structures. Data Structures in Java for the Principled Programmer. Capítulos 12 y 14.

---

<sup>1</sup> La versión más completa de un BST esta descrita en el libro de texto Java Structures, en el capítulo 14. Para esta hoja de trabajo no se requiere toda esa implementación.