***Seguimiento 11***

| TAD SearchBinaryTree |
| --- |
| SearchBinaryTree = {Node<T> node} |
| inv: |
| Operaciones Primitivas:   * isEmpty: →Boolean * showEmpty: SearchBinaryTree\_Node<T> root →void * getRoot: →SearchBinaryTree\_Node<T> * CreateTree: SearchBinaryTree\_Node<T> node, T data →void * search: SearchBinaryTree\_Node<T> node, T data →SearchBinaryTree\_Node<T> * insertNode: T data →SearchBinaryTree\_Node<T> * delete: SearchBinaryTree\_Node<T> node → void |
|  |

isEmpty: Funciona para verificar si root es nula.

showEmpty: Muestra un mensaje indicando cuando root es nulo.

getRoot: Obtiene la raiz(root)

CreateTree: Crea un arbol binario.

search: Busca el elemento dado en el nodo que se le da.

insertNode: Inserta un nuevo nodo en el arbol.

delete: Elimina un nodo en el arbol.