



## Práctica 7: Recorrido de árboles binarios

1. Dado el siguiente tipo de dato:

```
- datatype int newarbolbin =  
    Vacio |  
    Nodo of int newarbolbin * int * int newarbolbin;
```

- a. Defina la función `preOrden2` que realice el recorrido *preorden* de un `newarbolbin`
  - b. Defina la función `inOrden2` que realice el recorrido *ineorden* de un `newarbolbin`
  - c. Defina la función `postOrden2` que realice el recorrido *postorden* de un `newarbolbin`
2. Suponga que existe un diccionario implementado como un árbol binario de búsqueda de tipo `(string, string) newarbolbin`. Cada nodo del árbol es una tupla que contiene un término en español (escrito todo en minúscula y sin acentos) y una traducción al inglés (también escrita en minúscula). El árbol binario de búsqueda está ordenado de acuerdo al orden alfabético del término en español.
- a. Defina una función `listar_orden_alfabético` que tome el diccionario y devuelva un `(string, string) list` de todos los términos en español del diccionario junto a su traducción, ordenados de orden creciente de acuerdo al término en español. Debe implementarlo usando alguna de las funciones propuestas en el ejercicio 1.
  - b. Defina una función `buscar_letra` que tome un diccionario y una letra y devuelva un `(string, string) list` de todos los términos en español del diccionario que comienzan por dicha letra junto a su traducción, ordenados de orden creciente de acuerdo al término en español. Debe usar la función `listar_orden_alfabético` y `filter` para su definición.
3. Dado un árbol de tipo `int newarbolbin` Implemente las siguientes funciones usando lo que se le solicita:
- a. Implemente la función `suma` que sume todos los nodos del árbol. Debe usar `preOrden2` y `foldr`.
  - b. Implemente la función `producto` que multiplique todos los nodos del árbol. Debe usar `inOrden2` y `foldl`.