Задание № 3

Определим новый тип данных «[префиксное дерево](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%84%D0%B8%D0%BA%D1%81%D0%BD%D0%BE%D0%B5_%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE)» (Trie, бор). Словарь на основе префиксного дерева содержит пары (ключ, значение), где ключом является слово, а значением - «словарная статья» - объект некоторого типа. Каждый узел дерева представляет собой пару из словарной статьи (её может и не быть!) и списка дуг, помеченных символом и узлом-потомком.  
newtype Node a = Node (Maybe a, [Arc a])  
newtype Arc a = Arc (Char, Node a)  
Словарь, содержащий три слова “a”, “b” и “be” (соответствующие им словарные стьти – это строки “word a”, “word b” и “word be”), может быть представлен следующим значением:  
dictionary = Node (Nothing, [Arc ('a', Node (Just "word a", [])),   
 Arc ('b', Node (Just "word b",   
 [Arc ('e', Node (Just "word be", []))]))])  
Пустое дерево будет представлено значением Node (Nothing, []).

Написать две функции: первая, getEntries :: Node a -> [(String, a)] должна выдавать список пар (слово, словарная статья), содержащихся в дереве. Вторая, add :: (String, a) -> Node a -> Node a, должна добавлять слово со словарной статьёй в дерево.