

CS119 – Activity 11

Consider the abstract data type Bag:

Function	Explanation
<code>makeBag :: [a] -> Bag a</code>	convert a list to a bag.
<code>isEmpty :: Bag a -> Bool</code>	determines if a bag is empty
<code>union :: Bag a -> Bag a -> Bag a</code>	union of two bags
<code>minBag :: Bag a -> a</code>	returns the minimum value in the bag
<code>deleteMin :: Bag a -> Bag a</code>	removes one occurrence of the min value in the bag

Suppose that we decide to implement this ADT with a heap as described in the [notes](#) . Remember that in a heap the smallest value is the root and the left and right subtrees are also heaps.

Draw two or three different heaps for the multiset (or bag) $\{5,10,10,22,26,26,30\}$.

What happens for each of your trees when we `deleteMin`? Draw the new trees.