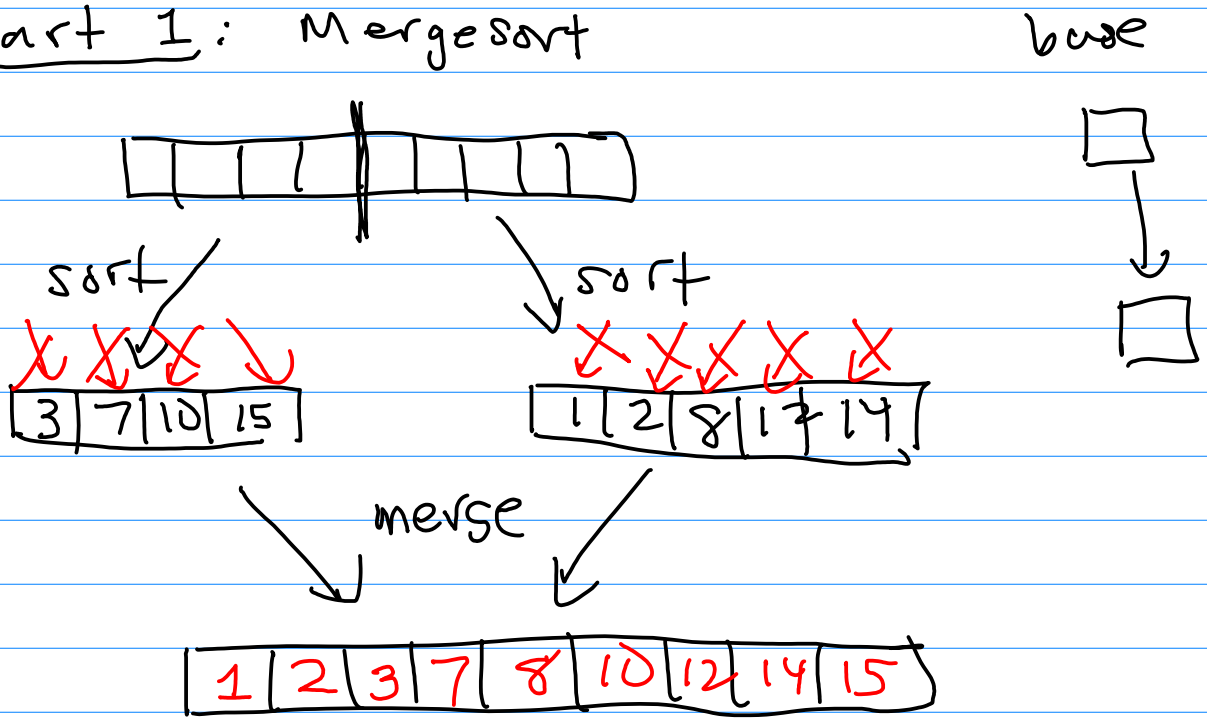
Part 1: Mergesort




```
ms :: Ord a => [a] -> [a]
merge :: Ord a => [a] -> [a] -> [a]
deal :: [a] -> ([a], [a])
deal [1,2,3,4,5] =
    ([1,3,5], [2,4])
```

[1,2,3]

[0,1,2,3]

[1,2]

[]

Snoc (Snoc (Snoc Nil 1) 2) 3

1 : 2 : 3 : [] → [1,2,3]

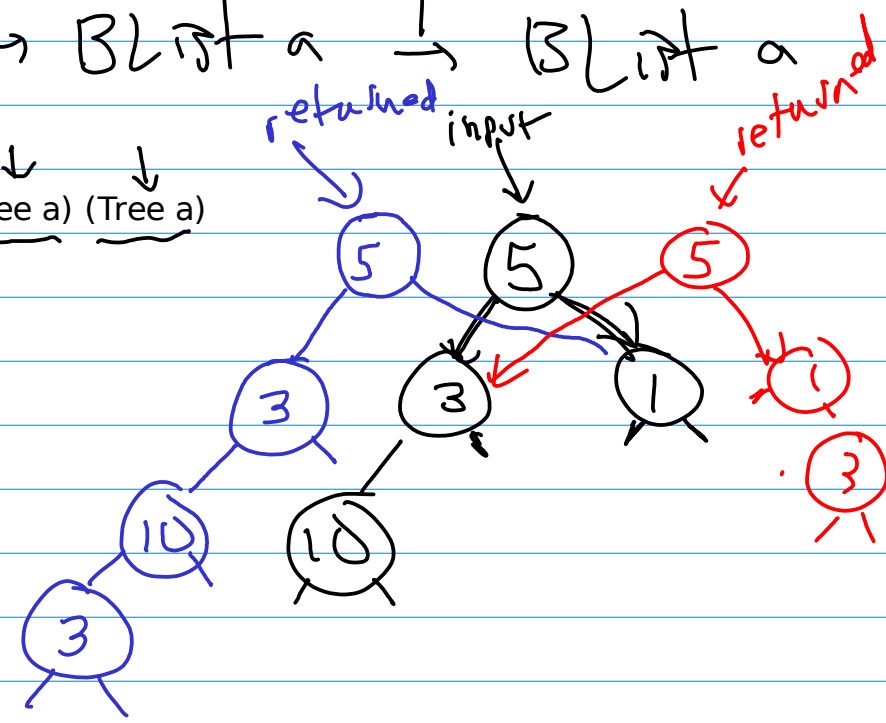
Cons 0 (Cons 1 (Cons 2 (Cons 3 Nil)))

(cons 4 Nil)

Snoc (Snoc (Snoc (Snoc Nil 0) 1) 2) 3

cons :: a → BList a → BList a

data Tree a = Empty | Node a (Tree a) (Tree a)



insert_left

insert_right

Persistent

Node 5 (Node 3 (Node 10 Empty Empty) Empty)

(Node 1 Empty Empty)

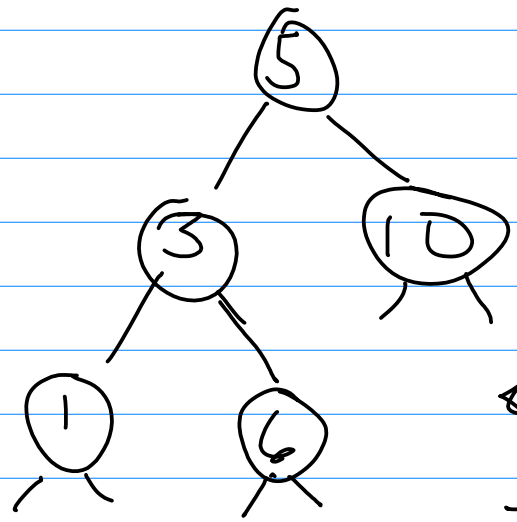
inorder

= [10, 3, 5, 1]

3

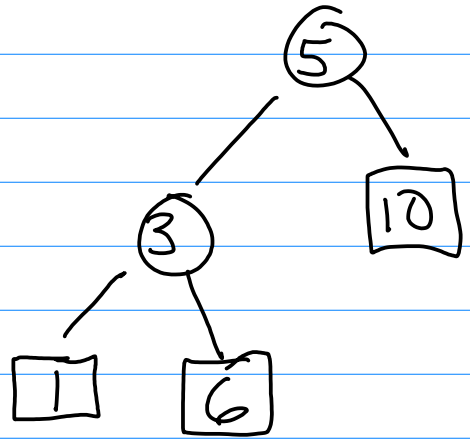
3

Tree a



' = Empty
○ = Node

Tree2 a



□ = Leaf
○ = Node