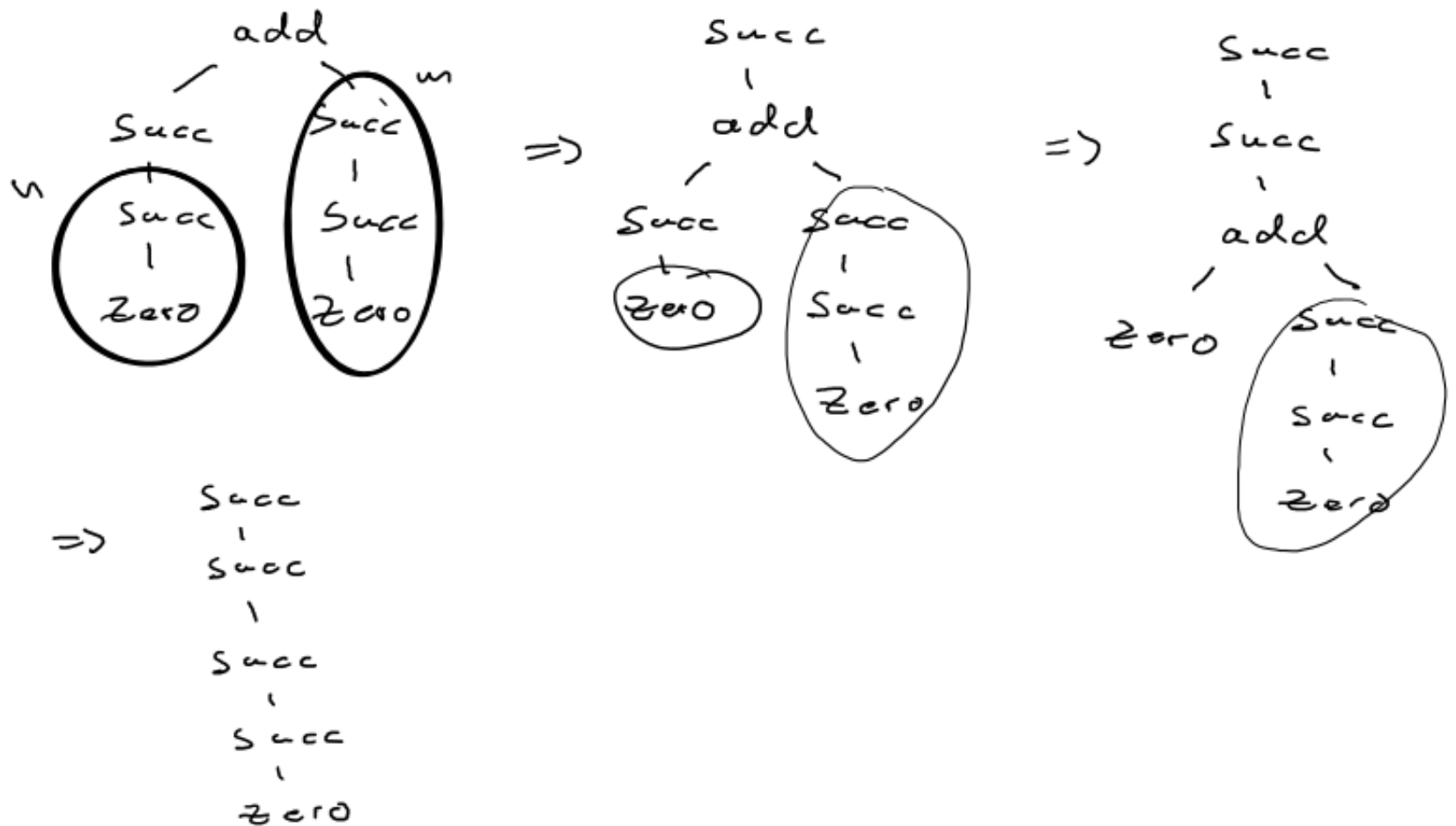


```

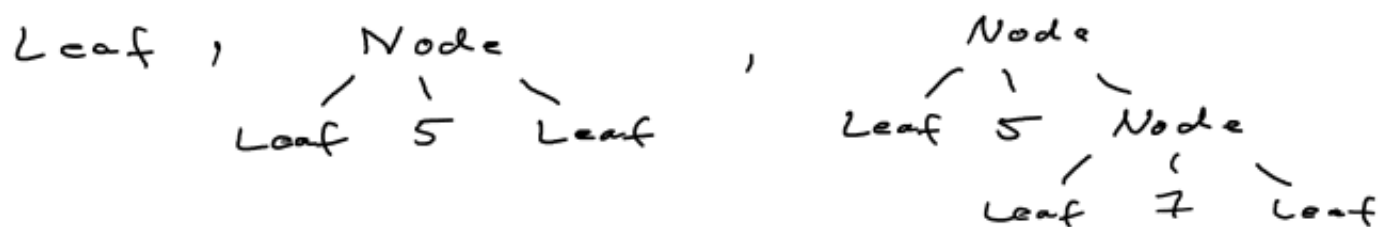
add :: Nat -> Nat -> Nat
add Zero      m = m
add (Succ n) m = Succ (add n m)

```



data Tree a = ... (as on slide)

some values of type Tree Integer:



some values of type Maybe (Tree Bool):

