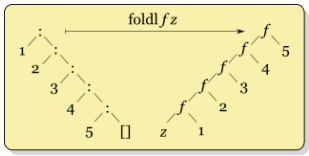
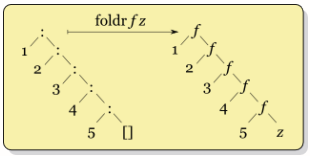
General libraries

Folding





If you operating over potentially infinite structures whilst building another structure, then you probably want a right associative fold (**foldr**).

Otherwise, if you are reducing to a single value, then you probably want a strict left fold (**foldl'**).

```
foldl :: Foldable t
      => (a -> b -> a) -> a -> t b -> a
foldr :: Foldable t
      => (a -> b -> b) -> b -> t a -> b

toList :: Foldable t => t a -> [a]

and, or       :: Foldable t
              => t Bool -> Bool
any, all      :: Foldable t
              => (a -> Bool) -> t a -> Bool
sum, product  :: (Foldable t, Num a)
              => t a -> a
minimum, maximum     :: (Foldable t, Ord a)
                     => t a -> a
minimumBy, maximumBy :: Foldable t
                     => (a -> a -> Ordering)
                     -> t a -> a

elem :: (Foldable t, Eq a)
     => a -> t a -> Bool
find :: Foldable t
     => (a -> Bool) -> t a -> Maybe a
```

```
> foldl' (flip (:)) [0] [1,2,3]
[3,2,1,0]

> foldr (:) [5] [1,2,3,4]
[1,2,3,4,5]

> take 5 $ foldr (:) [] [1..]
[1,2,3,4,5]
```