$$f(n) = \begin{cases} 0 & \text{if } n = 0 \quad 1 \\ 2n - 1 + f(n-1) & \text{if } n > 0 \quad 2 \end{cases}$$

$f(n) = n^2$ $(n \in \mathbb{N})$

for all

Base Case $(n=0)$

WTP) $f(0) = 0^2$

$$\begin{aligned} f(0) &= 0 && (1) \\ &= 0^2 \quad \square \end{aligned}$$

Inductive Case $(n=k+1)$

I.H) $f(k) = k^2$

WTP) $f(k+1) = (k+1)^2$

$$\begin{aligned} f(k+1) &= 2(k+1) - 1 + f(k) && (2) \\ &= 2k + 2 - 1 + f(k) \\ &= 2k + 1 + f(k) \\ &= 2k + 1 + k^2 && (\text{I.H}) \\ &= k^2 + 2k + 1 \\ &= (k+1)^2 \quad \square \end{aligned}$$

```haskell
sum :: [Int] -> Int
sum []     = 0            --1
sum (x:xs) = x + sum xs   --2

foldr :: (a -> b -> b) -> b -> [a] -> b
foldr f z []     = z                    --A
foldr f z (x:xs) = x `f` foldr f z xs   --B
```

sum ls = foldr (+) 0 ls

GOAL ↑

Base Case (ls = [])

WTP) sum [] = 0 ①

= 0

= foldr (+) 0 [] A^R

□

Inductive Case (ls = x:xs)

I.H) sum xs = foldr (+) 0 xs ←

WTP) sum (x:xs) = foldr (+) 0 (x:xs)

sum (x:xs) = x + $\boxed{\text{sum } xs}$ ②

= x + foldr (+) 0 xs I.H

= foldr (+) 0 (x:xs) B^R

□