4: LISTS

"There are two ways of constructing a software design: one way is to make it so simple that there are obviously no deficiencies and the other way is to make it so complicated that there are no obvious deficiencies. The first method is far more difficult.

C.A.R. Hoare, 1980.

List Anatomy

$$xs = [\overbrace{x_1}^{\text{head } xs}, \overbrace{x_2, \ldots, x_{n-1}, x_n}^{\text{tail } xs}]$$

$$xs = [\underbrace{x_1, x_2, \ldots, x_{n-1}}_{\text{init } xs}, \underbrace{x_n}_{\text{last } xs}]$$

(x:) Cons

(++[z]) snoc

List a   Empty   Cons

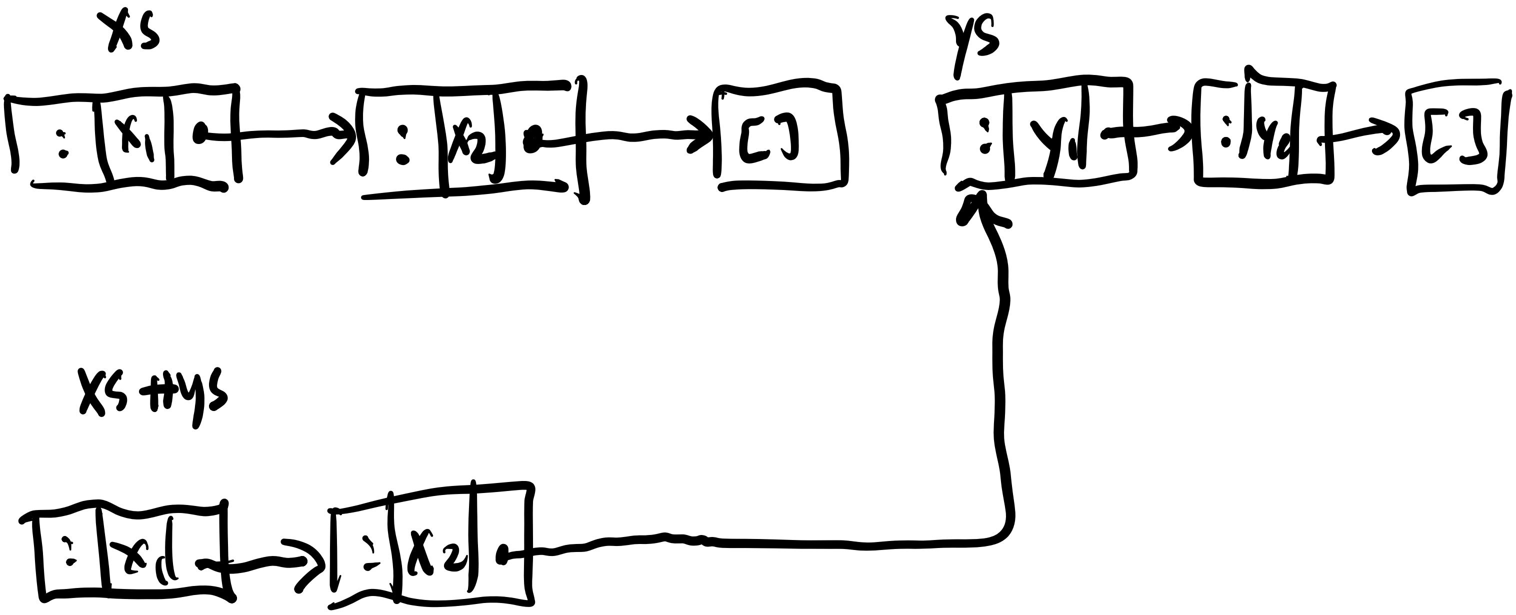
```
> data [a] = [] | (:) a [a]
```

or:

```
> data List a where
>       Empty :: List a
>       Cons  :: a -> List a -> List a
```

Lists are a <u>persistant</u> data structure

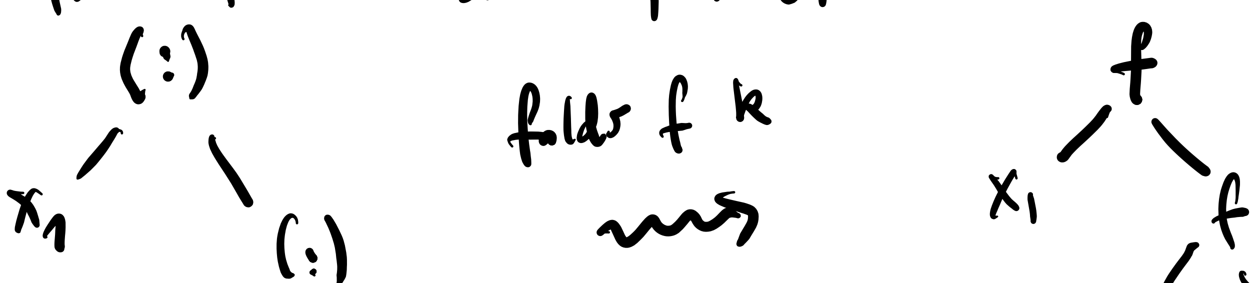
We can append lists with ++ :

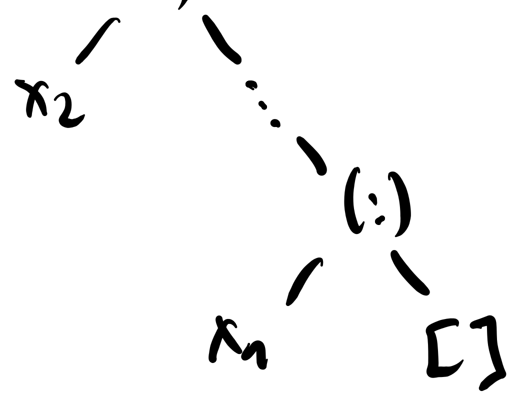


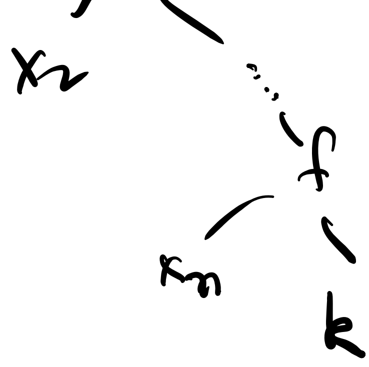

```
> (++) :: [a] -> [a] -> [a]
> [] ++ ys = ys
> (x:xs) ++ ys = x : (xs ++ ys)
```

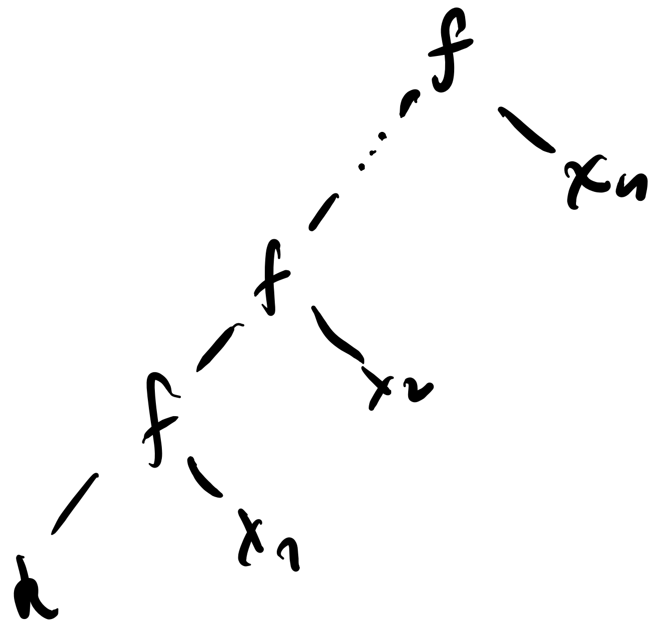
$$T(++)\ xs\ ys \in O(m)$$
where $m = \text{length } xs$

```
> foldr :: (a -> b -> b) -> b -> [a] -> b
> foldr f k [] = k
> foldr f k (x:xs) = f x (foldr f k xs)
```









$$\text{foldl} :: (b \to a \to b) \to b \to [a] \to b$$

$$\text{foldr } (:)\ [\,] = \text{id}$$
$$\text{foldl } (\text{snoc})\ [\,] = \text{id}$$

When is foldr = foldl?

Monoid: $(a, \diamond, \varepsilon)$ $\quad (\diamond) :: a \to a \to a$, $\varepsilon :: a$

$$x \diamond (y \diamond z) = (x \diamond y) \diamond z \quad \text{(associativity)}$$
$$\varepsilon \diamond y = y \quad \text{(left unit)}$$
$$x \diamond \varepsilon = x \quad \text{(right unit)}$$

$$\text{foldr } (\diamond)\ \varepsilon = \text{foldl } (\diamond)\ \varepsilon$$

Examples of Monoids:

$([a], ++, [\,])$

$(xs ++ ys) ++ zs = xs ++ (ys ++ zs)$

$(\mathbb{N}, *, 1)$

$(\mathbb{N}, +, 0)$

$(\mathbb{B}, \wedge, True)$, $(\sqcap, \infty)$, $(\cap, X)$

$(\mathbb{B}, \vee, False)$, $(\sqcup, -\infty)$, $(\cup, \phi)$

...

$(a \to a, (\cdot), id)$

$(f \cdot g)\, x = f\, (g\, x)$

$((f \cdot g) \cdot h)\, x = f\, (g\, (h\, x))$

$= (f \cdot (g \cdot h))\, x$

```
> concat :: [[a]] -> [a]
```

$\text{concat } [xs_1, xs_2, xs_3] = xs_1 +\!\!+\ xs_2 +\!\!+\ xs_3$

```
> concat []       = []
> concat (xs:xss) = xs ++ concat xss
```

```
> concatr :: [[a]] -> [a]
> concatr = foldr (++) []
```

```
> concatl :: [[a]] -> [a]
> concatl = foldl (++) [].
```

$T(+\!\!+)\ xs\ ys \in O(n)$ where $n = \text{length } xs$

For simplicity assume

$n = \text{length } xs_1 = \text{length } xs_2 = \cdots = \text{length } xs_{m+1}$

$\text{concatr } [xs_1, xs_2, \ldots, xs_{m+1}]$

$=$

$$xs_1 \underset{n}{+\!\!+} (xs_2 \underset{n}{+\!\!+} (xs_3 \underset{n}{+\!\!+} \cdots \underset{n}{+\!\!+} (xs_m \underset{n}{+\!\!+} xs_{m+1}))))$$

$$O(mn)$$

$$\text{concatl } [xs_1, xs_2, \ldots, xs_{m+1}]$$

$$= \text{rcl}\,(((xs_1 \mathbin{+\!\!+} xs_2) \mathbin{+\!\!+} xs_3) \mathbin{+\!\!+} \cdots \mathbin{+\!\!+} xs_m) \mathbin{+\!\!+} xs_{m+1})$$

$$n \qquad 2n \qquad 3n \qquad \qquad mn$$

$$O(m^2 n)$$

How do we fix this?

Insight: $(\cdot)$ is associative.

$$xs_1 \mathbin{+\!\!+} xs_2 \mathbin{+\!\!+} xs_3$$

$$\rightsquigarrow$$

$$f\,xs_1 \cdot f\,xs_2 \cdot f\,xs_3$$

$$\rightsquigarrow$$

$$((xs_1 \mathbin{+\!\!+}) \cdot (xs_2 \mathbin{+\!\!+}) \cdot (xs_3 \mathbin{+\!\!+}))\ []$$

$=$

$$xs_1 ++ (xs_2 ++ (xs_3 ++ []))$$

Difference Lists, Hughes 1986.

, Cayley 1854

> data DList a = DList ([a] → [a])

> toList :: DList a → [a]
> toList (DList $\underbrace{fxs}_{[a] \to [a]}$) = fxs []

> fromList :: [a] → DList a
> fromList xs = DList $\underbrace{(xs ++)}_{[a] \to [a]}$

[]

ε :: DList a
ε = DList id

\#

$(\diamond) :: \text{DList } a \to \text{DList } a \to \text{DList } a$

$\text{DList } fxs \diamond \text{DList } fys$

$$= \text{DList } (\lambda \underbrace{zs \to fxs\ (fys\ (zs))}_{[a] \to [a]})$$

or

$$= \text{DList } (fxs \cdot fys)$$

$T(++)\ xs\ ys \in O(m)$ where $m = \text{length } xs$

|

$\overline{++}$

$xs\ \overline{++}\ ys \in O(m)$

$xs\ \overline{++}\ (ys\ \overline{++}\ zs)$?

$(xs\ \overline{++}\ ys)\ \overline{++}\ zs$

$f\ \bar{\bullet}\ g\ \bar{\bullet}\ h$

?