① 1) ~~f1 : [a] -> b~~

f1 ls a = map (\f -> f a) ls

2) f2 f Leaf = []
f2 f (Node l v r) = f2 f l ++ (f v : f2 f r)

3) qs [] = []
qs (x:xs) = (qs [a | a <- xs, a <= x]) ++
[x] ++
(qs [a | a <- xs, a > x])

4) f4 f g x = f x (g x)

5) f5 f g x = f (\a -> g a x) x

(2) 1) $(\lambda x.xx)(\lambda x.xx) = \Omega$

Can not be normilized, because each reduction step doesnt change term.

$\Omega \xrightarrow{\beta} \Omega$

2) $(\lambda x.xxx)(\lambda x.xxx)$

~~Can~~ Can not be normilized, because grows after ~~each~~ each β-reduction step.

$(\lambda x.xxx)(\lambda x.xxx) \xrightarrow{\beta}{}^{n} = (\lambda x.xxx)\ldots(\lambda x.xxx)$

$n \neq 2$

3)

(3) 1) Int isomorphic to Nat, and
Nat isomorphic to [()]
(n ⇔ list of length n)

2) 4) Either a (Either b (c,d))
isomorphic to
Either (c,d) (Either a b)

Because
Left x ⇔ Right (Left x)
Right (Left x) ⇔ Right (Right x)
Right (Right x) ⇔ Left x

5) (a → b, a → c) isomorphic to
(a, a) → (b, c)

Because
(f, g) ⇔ \(x, y) → (f x, g y)

3) We can think about List as a
mapping Nat → a. Also Nat ⇔ Int
⇒ List a isomorphic to Map Int a

2) Figured out only primitive example
of Either () (Tree a, a, Tree a)