Revelations about the Y-combinator

Examples:

Map

$map\ \_\ [\,]=[\,]$
$map\ f\ (x{:}xs) = f x : map\ f\ xs$ (Haskell)

Turn into one-liner: (Scheme)

$map\ f\ l = \text{if}\ (null\ l)\ l\ (cons\ (f\ (head\ l))\ (map\ f\ (tail\ l)))$

"$map \equiv \lambda f.\lambda l.\ ifthen\ (null\ l)\ l\ (cons\ (f\,(head\ l))\ (map\ f\ (tail\ l)))$" (Lambda calculus)

$F \equiv \lambda g.\lambda f.\lambda l.\ ifthen\ (null\ l)\ l\ (cons\ (f\,(head\ l))\ (g\ f\ (tail\ l)))$

$\boxed{map \triangleq YF}$

$Y \equiv \lambda f.((\lambda x. f(x\,x))(\lambda x. f(x\,x)))$

$YF \to_\beta F(YF)$ "YF is the fixed point of F"

$map\ f'\ l' \equiv YF\ f'\ l'$

$\to_\beta F(YF)\ f'\ l' \equiv F(map)\ f'\ l'$ $(\lambda f.\lambda l.\ ifthen\ (null\ l)\ l\ (cons\ (f\,(head\ l))\ (map\ f\ (tail\ l))))\ f'\ l'$

$\to_\beta \lambda f.\lambda l.\ ifthen\ (null\ l)\ l\ (cons\ (f'(head\ l'))\ (map\ f'\ (tail\ l')))$

$\to_\beta^* ifthen\ (null\ l')\ l'\ (cons\ (f'(head\ l'))\ (map\ f'\ (tail\ l')))$

Foldr

$foldr\ \_\ a\ [\,] = a$
$foldr\ f\ a\ (x{:}xs) = foldr\ f\ (f\ a\ x)\ xs$

one-liner:

$foldr\ f\ a\ l = ifthen\ (null\ l)\ a\ (foldr\ f\ (f\ a\ (head\ l))\ (tail\ l))$

"$foldr \equiv \lambda f.\lambda a.\lambda l.\ ifthen\ (null\ l)\ a\ (foldr\ f\ (f\ a\ (head\ l))\ (tail\ l))$"

$F \equiv \lambda g.\lambda f.\lambda a.\lambda l.\ ifthen\ (null\ l)\ a\ (g\ f\ (f\ a\ (head\ l))\ (tail\ l))$

$foldr \triangleq YF$

Infinite Lists

"$buildList \equiv \lambda h.\lambda f.\ cons\ h\ (buildList\ (f\ h)\ f)$"

$F \equiv \lambda b.\lambda h.\lambda f.\ cons\ h\ (b\ (f\ h)\ f)$

$buildList \triangleq YF$

$[0,1,2,\dots] \equiv buildList\ 0\ succ$
$odds \equiv buildList\ 1\ (+2)$

$evens \equiv buildList\ 0\ (+2)$
$pow2 \equiv buildList\ 1\ (*2)$
$primes \equiv buidList\ 2\ nextPrime$

$nextPrime\ n =$
if $all\ (not\ (divides(n+1)))[2,\dots,n]$
then $n+1$
else $nextPrime\ (n+1)$