Programming constructs in 2 calculus e function that takes a single argument  $\chi$ . Ma single argument multiple arguments f(x,y) = M $f: \chi \times (\lambda y. M)$ (fa)b "Currying" after mathematician Haskell Corry  $(\lambda x. \lambda y. x+y)$  3  $(\lambda y. 3+y)$  23+2=5

## Church Booleans

tru =  $\lambda t. 2f. t$  3 functions

fls =  $\lambda t. 2f. f$  3 that take two arguments

NOT =  $\lambda x. (\alpha fls) tru$ poply a to fk then tru

tru/fls

if a is he" it returns the first

argument fls

if x is "false" it redurns the second tru

conditional

if (b) then y else w

cond = 2l. 2m. 2n. (l m) n

cond tru v w

= (2l. 2m. 2n. l m n) tru v w

> (2m. 2n. tru m n) y w

tru v w

> tru v w

> tru v w

AND = 
$$\lambda p. \lambda q. (pq) p$$
  
 $(pq) \in \mathbb{I}s$   
 $OR = \lambda p. \lambda q. (pp) q$   
 $pair = \lambda f. \lambda s. \lambda b. b f s$   
 $fst = \lambda p. p tru$   
 $Snd = \lambda p. p f ls$   
 $create(V, w)$   
 $pair v w \rightarrow \lambda b. b v w$   
 $fst (pair v w)$   
 $fst (\lambda b. b v w)$   
 $= (\lambda p. p tru) (\lambda b. b v w)$   
 $\rightarrow (\lambda b. b v w)$   
 $\rightarrow (\lambda b. b v w)$ 

Church Numerals

$$C_0 = \lambda s. \lambda z. z$$
 $\lambda t. z$ 
 $\lambda t. z$ 

Proc = 
$$\lambda n$$
.  $\lambda s$ .  $\lambda \pm$ .  $s$ .  $n$   $s$   $t$ .

Temove lambdas

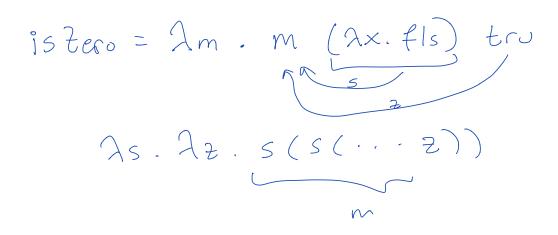
instat

add

 $\lambda s$  back

inc 
$$(\lambda s', \lambda t', t')$$
  
 $\rightarrow \lambda s, \lambda t, s$   $((\lambda s', \lambda t', t') s t)$   
 $\rightarrow \lambda s, \lambda t, s$   $t$ 

Homes =  $\lambda m$ .  $\lambda n$ . m (plus n) C.  $\lambda z$ .  $S \leq S \leq S \leq N$ .  $\lambda z$ .  $S \leq S \leq S \leq N$ .  $\lambda \leq S \leq S \leq N$ .



Z (As, Az. Z) (Ax. Fls) tru (As, Az. SZ) (Ax. Fls) tru (As, Az. SZ) (Ax. Fls) tru (Ax. Fls) tru (Ax. Fls) tru Exponentiation

 $M^{n}$ 

exp = 2 m. 2 n. n m