Hunes treferring—We give hames to functions so the can use it more than
Functions so the can use it more than
once (-> Lambda + definite integrals).
,
Recursion is all solving equations.
Don't need the gradratic equation,
Don't herd the gradratic equation, you just need the fixed-point.
Patt: Problem 3 — Think of random choice as subjuting a procedure (strategy). Higher-order procedures.
as substituting a procedure (strategy).
Higher-order procedures.
•
Inputs a Shid-state (randomly
chrises 1 of 2 strategics), a procedure,
Inputs a Ship-state (randomly choises 1 of 2 strategies), a procedure, and outputs another procedure/strate-
$au\cdot$
random-choice: (strategy * strategy)
•
strategy
Problem #4
height-choice: (strategy *strategy* int)
Strategy
#50hoice: (strategy strategy to shipstate > n
ENOICE ISTUTERY STYCHEDY ISKIPSTANC 7

strategy
Curry
* Programs are types proofs, & types are theorems. f: A > B } f is a proad re turning input of type A Into output of type B (or) f is a proof of theorem A > B
finales.
of the A lets metout of the B
(or) f is a proof of theorem A -> B
'
x:A
Modus ponuns
A->A: (7(x)x) } the identity fxn
* Lambda expressions evaluate to procedures,