Syntactic Transformation To Monadic Form

- ***Expressions:***

------ exp -------

$desugar_{<exp>}$ *:: Exp → Exp*

$desugar_{<exp>}$ exp = $desugar_{<lexp>}$ exp return h

------- lexp -------

$desugar_{<lexp>}$ *:: Exp → Exp*

-------lexp: fexp -------

$desugar_{<lexp>}$ fexp = $desugar_{<fexp>}$ fexp

ΠΡΟΒΛΗΜΑ ΑΝ έχει 1 argument

------- fexp -------

$desugar_{<fexp>}$ aexp = $desugar_{<aexp>}$ aexp

$desugar_{<fexp>}$ (fexp literal) = $desugar_{<lexp>}$ fexp literal >>= \h →

$desugar_{<fexp>}$ (fexp qvar) = $desugar_{<lexp>}$ fexp qvar >>= \h →

$desugar_{<fexp>}$ (fexp gcon) = $desugar_{<lexp>}$ fexp gcon >>= \h →

$desugar_{<fexp>}$ (fexp (exp))= exp >>= \ex_i → $desugar_{<lexp>}$ fexp ex_i >>= \h →

------- aexp -------

$desugar_{<aexp>}$ literal = literal

$desugar_{<aexp>}$ qvar = qvar

$desugar_{<aexp>}$ gcon = gcon

$desugar_{<aexp>}$ (exp) = $desugar_{<lexp>}$ exp

-------lexp: let decls in exp -------

$desugar_{<lexp>}$ (let decls in exp) = $desugar_{<dclrs>}$ decls exp >>= \h ->

- ***Declarations***

------- dclrs -------

desugar$_{\texttt{<dclrs>}}$ *::* *Dclrs → Exp*

desugar$_{\texttt{<dclrs>}}$ (dclr1; … ; dclrn) = desugar$_{\texttt{<dclr>}}$ declr1 … desugar$_{\texttt{<dclr>}}$ declrn
| (;) = _ ->

------- dclr -------

desugar$_{\texttt{<dclr>}}$ (funlhs | pat) (= exp) = exp >>= \(funlhs | pat) ->