Contents

1 exType Theory

Built: 15 February 2020
Parent Theories: indexedLists, patternMatches

1.1 Definitions

[APP_def]

$\vdash$ $(\forall l.$ APP [] l = $l) \wedge \forall h\ l_1\ l_2.$ APP $(h::l_1)\ l_2$ = $h::$APP $l_1\ l_2$

[Map_def]

$\vdash$ $(\forall f.$ Map f [] = []$) \wedge \forall f\ h\ l_1.$ Map f $(h::l_1)$ = $f\ h::$Map $f\ l_1$

1.2 Theorems

[APP_ASSOC]

$\vdash \forall l_1\ l_2\ l_3.$ APP (APP $l_1\ l_2$) l_3 = APP l_1 (APP $l_2\ l_3$)

[LENGTH_APP]

$\vdash \forall l_1\ l_2.$ LENGTH (APP $l_1\ l_2$) = LENGTH l_1 + LENGTH l_2

[Map_APP]

$\vdash$ Map f (APP $l_1\ l_2$) = APP (Map $f\ l_1$) (Map $f\ l_2$)

2 nexp Theory

Built: 15 February 2020
Parent Theories: indexedLists, patternMatches

2.1 Datatypes

nexp = Num num | Add nexp nexp | Sub nexp nexp | Mult nexp nexp

2.2 Definitions

[nexpVal_def]

$\vdash$ $(\forall num.$ nexpVal (Num num) = $num) \wedge$
$(\forall n_1\ n_2.$ nexpVal (Add $n_1\ n_2$) = nexpVal n_1 + nexpVal $n_2) \wedge$
$(\forall n_1\ n_2.$ nexpVal (Sub $n_1\ n_2$) = nexpVal n_1 - nexpVal $n_2) \wedge$
$\forall n_1\ n_2.$ nexpVal (Mult $n_1\ n_2$) = nexpVal $n_1 \times$ nexpVal n_2

2.3 Theorems

[Add_0]

$\vdash \forall n.$ `nexpVal (Add (Num 0) n) = nexpVal n`

[Add_SYM]

$\vdash \forall n_1\ n_2.$ `nexpVal (Add` n_1 n_2`) = nexpVal (Add` n_2 n_1`)`

[Mult_ASSOC]

$\vdash \forall n_1\ n_2\ n_3.$
`nexpVal (Mult` n_1 `(Mult` n_2 n_3`)) =`
`nexpVal (Mult (Mult` n_1 n_2`)` n_3`)`

[Sub_0]

$\vdash \forall n.$
`(nexpVal (Sub (Num 0)` n`) = 0)` $\wedge$
`(nexpVal (Sub` n `(Num 0)) = nexpVal` n`)`

Index