\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* NOMBRE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

fmod NATURAL is

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CONJUNTOS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

protecting BOOL .

sort N .

sort NoN .

subsort NoN < N .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SINTAXIS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

op cero : -> N .

op sucesor : N -> N .

op suma : N N -> N .

op esCero : N -> Bool .

op esIgual : N N -> Bool .

op esDistinto : N N -> Bool .

op NODEFINIDO : -> NoN .

op INFINITO : -> NoN .

op NEGATIVO : -> NoN .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SEMANTICA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

var n m : N .

eq suma(cero, n) = n .

eq suma(sucesor(m), n) = sucesor(suma(m, n)) .

eq esCero(cero) = true .

eq esCero(sucesor(n)) = false .

eq esIgual(cero, n) = esCero(n) .

eq esIgual(sucesor(n), cero) = false .

eq esIgual(sucesor(n), sucesor(m)) = esIgual(n, m) .

eq esDistinto(n, m) = not esIgual(n, m) .

endfm

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* NOMBRE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

fmod VOCAL is

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CONJUNTOS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

protecting BOOL .

sort V .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SINTAXIS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ops A E I O U : -> V .

op esIgual : V V -> Bool .

op esDistinta : V V -> Bool .

op esMenor : V V -> Bool .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SEMANTICA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

var v w : V .

eq esIgual(v, v) = true .

eq esIgual(v, w) = false .

eq esDistinta(v, w) = not esIgual(v, w) .

eq esMenor(v, v) = false .

eq esMenor(A, E) = true .

eq esMenor(A, I) = true .

eq esMenor(A, O) = true .

eq esMenor(A, U) = true .

eq esMenor(E, I) = true .

eq esMenor(E, O) = true .

eq esMenor(E, U) = true .

eq esMenor(I, O) = true .

eq esMenor(I, U) = true .

eq esMenor(O, U) = true .

eq esMenor(U, O) = false .

eq esMenor(U, I) = false .

eq esMenor(U, E) = false .

eq esMenor(U, A) = false .

eq esMenor(O, I) = false .

eq esMenor(O, E) = false .

eq esMenor(O, A) = false .

eq esMenor(I, E) = false .

eq esMenor(I, A) = false .

eq esMenor(E, A) = false .

endfm

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* NOMBRE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

fmod LISTA is

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CONJUNTOS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

protecting BOOL .

protecting NATURAL .

protecting VOCAL .

sort L .

sort MensajeListas .

subsorts MensajeListas < V .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SINTAXIS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

op listaVacia : -> L .

op esVacia : L -> Bool .

op insertar : V L -> L .

op insertarFinal : V L -> L .

op tamano : L -> N .

op ErrorListaVacia : -> MensajeListas .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SEMANTICA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

var l : L .

var v w : V .

eq esVacia(listaVacia) = true .

eq esVacia(insertar(v, l)) = false .

eq insertarFinal(v, listaVacia) = insertar(v, listaVacia) .

eq insertarFinal(v, insertar(w, l)) = insertar(w, insertarFinal(v, l)) .

eq tamano(listaVacia) = cero .

eq tamano(insertar(v, l)) = sucesor(tamano(l)) .

endfm