TDA Lista

CLASS Lista [Elemento]

IMPORTS Boolean, Natural

BASIC CONSTRUCTORS CrearL, AgregarL

EFFECTIVE

TYPE

Lista

OPERATIOS

crearL: -> Lista;

cantElemL: Lista -> Natural;

perteneceL: Lista \* Elemento -> Boolean;

esVaciaL: Lista -> Boolean;

agregar: Lista \* Elemento -> Lista;

eliminarL: Lista(l) \* Elemento(e) -> Lista

pre: pertenceL(l,e);

AXIOMS …

.

.

.

END\_CLASS;

TDA Conjunto

CLASS Conjunto [Elemento]

IMPORTS Boolean, Natural

BASIC CONSTRUCTORS CrearC, AgregarC

EFFECTIVE

TYPE

Conjunto

OPERATIONS

crearC: -> Conjunto;

cantElemC: Conjunto -> Natural;

perteneceC: Conjunto \* Elemento -> Boolean;

agegarC: Conjunto \* Elemento -> Conjunto

eliminarC: Conjunto(c) \* Elemento(e) -> Conjunto

pre: pertenceC(c,e);

unionC: Conjunto \* Conjunto -> Conjunto;

interseccionC: Conjunto \* Conjunto -> Conjunto;

diferenciaC: Conjunto \* Conjunto -> Conjunto;

AXIOMS …

.

.

.

END\_CLASS

TDA ARBOL BINARIO

CLASS ArbolB [Elemento]

IMPORTS Boolean, Natural, Lista [Elemento]

BASIC CONSTRUCTORS CrearA, AgregarA

EFFECTIVE

TYPE

ArbolB

OPERATIOS

crearA: -> ArbolB;

agregarA: ArbolB \* ArbolB \* Elemento -> ArbolB;

perteneceA: ArbolB \* Elemento -> Boolean;

cantElemA: ArbolB -> Natural;

esVacioA: ArbolB -> Boolean;

profundidadA: ArbolB -> Natural;

fronteraA: ArbolB -> Lista;

mostrarInorder: ArbolB -> Lista;

AXIOMS …

.

.

.

END\_CLASS;

TDA LIBRO

CLASS libro

IMPORTS natural, cadena

BASIC CONSTRUCTORS creaL

EFFECTIVE TYPE libro

OPERATIONS

crearL : cadena\*cadena\* natural\*cadena\*cadena\*cadena ->libro;

obt\_T : libro → cadena;

obt\_A : libro → cadena;

obt\_P : libro → natural;

obt\_G : libro → cadena;

obt\_C : libro → cadena;

void mod\_T (string titulo);

void mod\_A (string autor);

void mod\_P (string paginas);

void mod\_E (string editorial);

void mod\_G (string genero);

void mod\_C (string contenido);

AXIOMS … … … …

.

.

.

END CLASS;

**TDA INDICE**

**CLASS** Indice

**IMPORTS** libro ,conjunto[libro]

**BASIC** **CONSTRUCTORS** crearIndice

**EFFECTIVE TYPE** Indice

**OPERATIONS**

crearIndice: → Indice;

agregarLibro : Indice \* libro → Indice;

devolverGenero : Indice \* cadena → conjunto[libro];

**AXIOMS** … … … …

.

.

.

**ENDO CLASS**;