TAD ListaDoble<a>

* Igualdad Observacional: Sean A y B dos listas dobles, son iguales si: la longitud de A y B son iguales, y cada elemento de A es igual al correspondiente de B.
* Usa: Natural, Bool, Secuencia<a>, None, coordenada< ListaDoble<a>>.
* Parámetro Formal: a.
* Género: ListaDoble<a>.
* Observadores Básicos:
  + - * Tamaño(ListaDoble<a>)🡪Natural.
      * Es\_vacia(ListaDoble<a>)🡪Bool.
      * Primero(ListaDoble<a>)🡪a. Pre{La lista doble tiene al menos un elemento}
      * Ultimo(ListaDoble<a>)🡪a. Pre{La lista doble tiene al menos un elemento}
* Generadores:
  + - * Vacía()🡪ListaDoble<a>. Post{La lista doble retornada esta vacia}
      * A\_partir\_de(Secuencia<a>)🡪 ListaDoble<a>. Post{La lista doble retornada contiene los elementos de la secuencia con el mismo orden que tiene la secuencia}
* Otras Operaciones:
  + - * Agregar\_frente(ListaDoble<a>, a)🡪None. Post{La lista doble contiene al menos un elemento}
      * Agregar\_Final(ListaDoble<a>, a)🡪None. Post{La lista doble contiene al menos un elemento}
      * Insertar(ListaDoble<a>,coordenada< ListaDoble<a>>, a)🡪None. Pre{La coordenada es válida}. Post{La lista doble no está vacia}
      * Borrar(ListaDoble<a>, coordenada< ListaDoble<a>>)🡪 Coordenada<ListaDoble<a>>. Pre{La coordenada es válida}
      * Inicio(ListaDoble<a>)🡪 Coordenada<ListaDoble<a>>.
      * Fin(ListaDoble<a>)🡪 Coordenada<ListaDoble<a>>.
* Axiomas:
  + - * Vacía()🡪Crea una lista doble sin elementos.
      * A\_partir\_de(Secuencia<a> S)🡪Crea una lista doble con los elementos de la secuencia S recibida.
      * Insertar(ListaDoble<a> L, coordenada< ListaDoble<a>> C, a ) 🡪 Agrega el elemento “a” en L en la posición que hace referencia la coordenada C.
      * Borrar(ListaDoble<a> L, coordenada< ListaDoble<a>> C)🡪Borra de la lista doble el elemento que se encuentra en la posición que hace referencia la coordenada C. Retorna la coordenada siguiente a C.
      * Tamaño(ListaDoble<a> L)🡪Devuelve la cantidad de elementos que posee la lista doble L.
      * Es\_vacia(ListaDoble<a> L)🡪Devuelve Verdadero si L esta vacia, o Falso si L no está vacia.
      * Primero(ListaDoble<a> L)🡪Retorna el primer elemento de la lista doble L.
      * Ultimo(ListaDoble<a> L)🡪Devuelve el último elemento de la lista doble L.
      * Agregar\_frente(ListaDoble<a> L, a)🡪Agrega el elemento “a” al frente de L.
      * Agregar\_Final(ListaDoble<a> L, a)🡪Agrega el elemento “a” al final de L.
      * Inicio(ListaDoble<a> L)🡪Retorna una coordenada que hace referencia al primer elemento de L.
      * Fin(ListaDoble<a> L)🡪Retorna una coordenada que hace referencia al último elemento de L.
* Exporta:
  + - * ListaDoble<a>, Tamaño, Es\_vacia, Primero, Ultimo, Vacía, A\_partir\_de, Agregar\_frente, Agregar\_Final, Insertar, Borrar, Inicio, Fin)