

Práctico 7 – TAD Lista (avanzado)

OBJETIVOS:

- Profundizar en algoritmia más avanzada de listas

Ejercicio 1

Implemente las siguientes funciones para la estructura de **lista simplemente encadenada**.

- a) **int maximo ();**
Pre: La lista no es vacía.
Pos: Retorna el máximo elemento de la lista.
- b) **int promedio ();**
Pre: La lista no es vacía.
Pos: Retorna el promedio de los elementos de la lista.
- c) **int contar (int elem);**
Pos: Retorna la cantidad de veces que aparece el elemento pasado como parámetro en la lista.
- d) **Lista cambiar (int n, int m);**
Pos: Retorna la lista resultado de cambiar todas las ocurrencias de n por el elemento m.
- e) **void insertarOrdenado (int elem);**
Pre: La lista está ordenada ascendentemente.
Pos: Inserta el elemento pasado como parámetro de forma ordenada en la lista.
- f) **Lista invertir ();**
Pos: Retorna una nueva lista, resultado de invertir el orden de los elementos de la lista original.
- g) **boolean estaOrdenada ();**
Pos: Retorna true sii la lista está ordenada.

Ejercicio adicionales

Implemente las siguientes funciones para la estructura de **lista simplemente encadenada**.

- a) **boolean esIgual (Lista p);**
Pos: Retorna true si la lista es igual a la lista p. Dos listas son iguales si son vacías o si tienen los mismos elementos y en el mismo orden.
- b) **Lista intercalar (Lista p);**
Pre: Las listas están ordenadas.
Pos: Retorna una nueva lista resultado de intercalar ordenadamente los elementos de p y

actual.

c) Lista concatenar (Lista p);

Pos: Retorna una nueva lista, resultado de concatenar la list p al final de la lista actual.

d) boolean estaIncluida (Lista p);

Pos: Retorna true si una lista está incluida en una lista p (si es posible encontrar una sub---
secuencia de elementos de p igual a la secuencia de elementos de la lista).