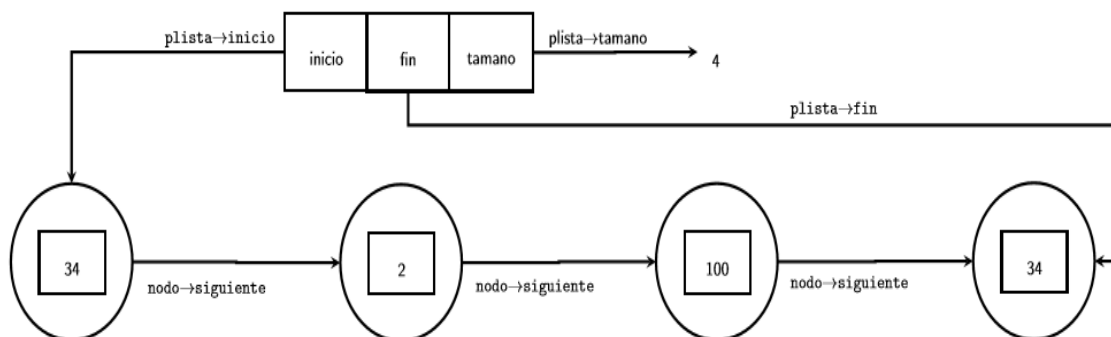


## Listados de Ejercicios para Estudio Solemne 1

- 1) Escriba una función: **moverSgteAlFrente(lis, pos)**. Esta función permite mover el nodo siguiente apuntado por **pos** al comienzo de la lista **lis**, donde **lis** es un puntero a la cabecera de una lista enlazada simple y **pos** es un puntero a un nodo de la lista.
- 2) Escriba una función: **compararPilas(pila1, pila2)** la cual devuelve verdadero si las pilas pila1 y pila2 son iguales, en caso contrario retorna falso. Utilice las funciones elementales **tope(pila)**, **pop(pila)**, **push(x, pila)** y **vacía(pila)**.
- 3) Escriba una función que reciba como argumento un número entero, lo separe en los dígitos que lo componen, y los guarde dichos dígitos en una lista. La función debe retornar la lista con los dígitos.
- 4) Realizar una función que reciba dos listas y que muestre todos los elementos de la primera lista que se encuentren en la segunda lista.
- 5) Escriba un programa que lea una cantidad par de caracteres. Después el programa muestra el primer carácter, el tercer carácter, el quinto carácter, etc. En una segunda línea se muestra el segundo carácter, el cuarto carácter, el sexto carácter, etc. Use dos colas para almacenar los caracteres.
- 6) Escriba una función que invierta una lista utilizando una estructura de datos auxiliar pila. La función puede ejecutar funciones elementales de una pila.
- 7) Escriba una función que calcula la longitud de una cola, es decir determina la cantidad de elementos que tiene la cola. La función debe utilizar sólo llamadas a tipos de fundamentales de cola. Este ejercicio supone que NO existe la columna tamaño en el typedef Cola.
- 8) Dada una lista no ordenada de números naturales, se solicita implementar una función que recibiendo una lista **L** y dos valores enteros **x**, **d**; devuelva el elemento que se encuentra a la distancia **d** del nodo que contiene el elemento **x**. Defina la estructura indicada en el ejemplo.

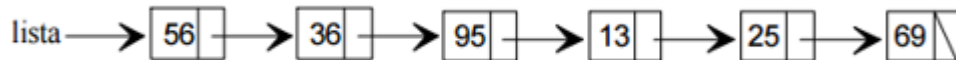


Según el ejemplo de la figura, para **x = 2** y **d = 2**, el método deberá devolver el valor **34**, que es el elemento que se encuentra a una distancia **d** (2 posiciones) del nodo que contiene el elemento **x** (con valor 2).

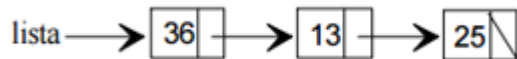
**Observaciones:** Si el elemento **x** no se encuentra en la lista, el método devolverá el valor 0. Igualmente, si no existe el elemento **d** posiciones después del elemento **x**, se sustituirá su valor por 0. Suponer que no existen elementos repetidos y que **d** es mayor que cero.

- 9) Cómo quedaría su implementación anterior si la lista es circular. Implemente este caso.
- 10) Mediante el uso de pilas verifique si los paréntesis de una función están balanceados, es decir, que a cada paréntesis abierto le corresponde un paréntesis cerrado. Considere que la función llega como parámetro, y es un arreglo dinámico de largo n.
- 11) Escribir una función que reciba como argumento dos colas, las mezcle y retorne una única cola como resultado. Los datos en la cola de resultado deben quedar ordenados.

- 12) Si tiene una lista enlazada que almacena números enteros. Implemente una función en C que suprima de la lista todos los elementos mayores que un número dado como límite. Por ejemplo, si la lista inicial es:



y el número dado como límite es 43, la lista resultante es:



- 13) Dada una cola C de enteros, retornar una nueva cola que contiene a todos los elementos de C (en el mismo lugar de la cola original), incrementados en 1. Si C es una cola vacía, la función debe retornar la cola vacía.
- 14) Realizar una función que elimine los elementos repetidos de una pila dada. En la figura se muestra el estado de la pila antes y después de la llamada a la función.

Pila P antes	Pila P después
3	
4	
3	
4	3
5	4
6	5
3	6

- 15) Una forma de implementar una cola circular C es hacer que el puntero C apunte al final de la cola y el siguiente nodo es el frente de la cola (como se muestra en la figura). Implemente una función la cual inserte un elemento al final de la cola circular. El elemento insertado no debe estar repetido, por lo tanto, debe retornar un verdadero si se realizó la inserción o falso en caso contrario.

