

## Comentarios generales

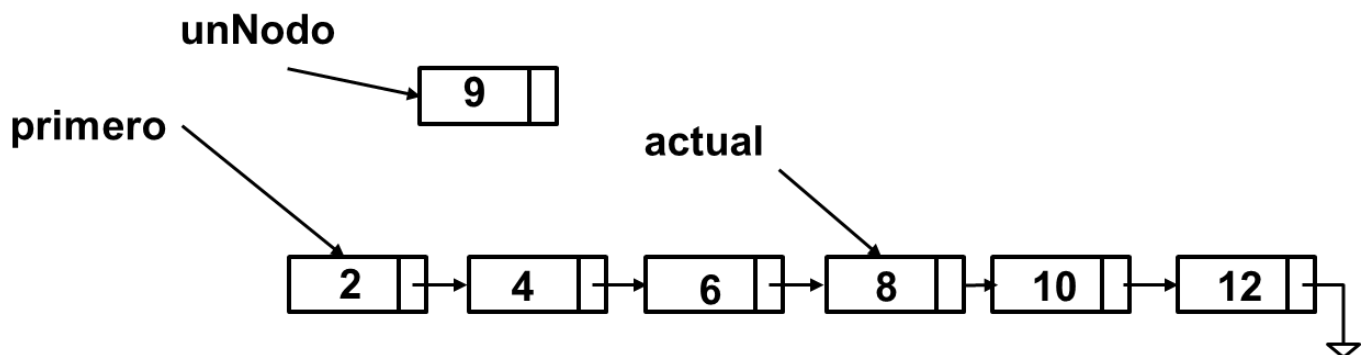
No confundir la implementación del método de lista, con el uso de ésta en diferentes contextos. El insertar de lista, inserta a partir de la etiqueta del nodo. Luego, de acuerdo al problema en el que se está usando la lista, la etiqueta del nodo se instanciará con el atributo del dato que interese (cédula, nombre, título, código, ...)

La decisión de no insertar etiquetas existentes, no puede estar en la lista, sino en el contexto del problema. Si quisiéramos, por ejemplo, ordenar una lista de personas por apellido, no queríamos que “filtrara” todos los apellidos iguales. Por lo tanto, los métodos de inserción en la lista no deben controlar eventuales etiquetas “duplicadas”

Mantener el pseudocódigo lo más sencillo e intuitivo posible, sin elementos de ningún lenguaje en particular. No usar por ejemplo “compareTo”.

## Recomendaciones para elaborar una solución

Siempre comenzar conceptualizando a partir del dibujo del modelo, y “correr” a mano el algoritmo, la forma más segura para evitar errores:



## Posible pseudo código detallado

insertaOrdenado (TNode unNode)

COM

Si unNode == nulo entonces devolver falso fin si      **O(1)**

Si primero == nulo entonces      **O(1)**

    unNode.siguiente = nulo      **O(1)**

    primero = unNode      **O(1)**

    devolver true      **O(1)**

Fin si

Si unNode.etiqueta < primero.etiqueta entonces      **O(1)**

    unNode.siguiente = primero      **O(1)**

    primero = unNode      **O(1)**

    devolver true      **O(1)**

fin si

actual = primero      **O(1)**

mientras (actual.siguiente <>nulo) Y (actual.siguiente.etiqueta ≤ unNode.etiqueta) hacer      **O(N)**

    actual = actual.siguiente      **O(1)**

fin mientras

unNode.siguiente = actual.siguiente      **O(1)**

actual.siguiente = unNode      **O(1)**

devolver true      **O(1)**

FIN

## Orden del tiempo de ejecución

Es necesario poner el orden de cada una de las sentencias del algoritmo, y luego aplicando las reglas de la suma y del producto, hallar el orden del algoritmo.

En este algoritmo, todas las sentencias de selección y de asignación son de orden constante.

El bloque del bucle es de orden constante y el bucle se repite hasta tantas veces como la cantidad de nodos de la lista, sea  $N$ . Por lo tanto, la sentencia "mientras" es de orden lineal  $O(N)$ . Luego, por la regla de la suma, el algoritmo es  $O(N)$ .

Se podría precisar un poco más, ya que puede ser el más chico, con lo que se resuelve con una comparación, o el mayor, con lo que habría que hacer las  $N$  comparaciones. Si la probabilidad de que haya que insertar en cualquier posición  $i$  entre 1 y  $N$  es la misma e igual a  $1/N$ , se demuestra que la cantidad de comparaciones promedio que hay que hacer es  $(N+1)/2$ , o sea  $O(N)$ .