



# Programación Modular

Examen extraord. ETSI Telecomunicación

20 de diciembre, 2006

APELLIDOS, Nombre

Grupo

Nº Conv

Entregar esta hoja. Poner "Apellidos, Nombre", letra del Grupo y número de la convocatoria.

- 1 3pt Implementar la clase **CCadena** para almacenar cualquier longitud de cadena usando como atributo un objeto de la clase **CLista**. Las letras de la cadena se guardan en la lista. Se pueden tener los atributos adicionales que se consideren necesarios, pero han de ser de tipo simple.

A continuación se muestran los interfaces de ambas clases. La clase **CLista** se ha de usar a nivel de usuario y se ha de implementar la clase **CCadena**.

INTERFAZ CLASE **CCadena**

METODOS

```

Crear()
Destruir()
Subcadena(S CCadena s, E N pos, E N cuantos)
    deja en s los cuantos caracteres desde la pos
Insertar(E N pos, E CCadena s)
    inserta en la pos la cadena s
Insertar(E N pos, E char c)
    inserta en la pos el carácter c
Substituir(E N pos, E CCadena s)
    substituye a partir de la pos los caracteres
    por los de s
Substituir(E N pos, E char c)
    substituye en la pos el carácter por c
Borrar(E N pos, E N cuantos)
    borra en la pos cuantos caracteres
N Longitud()
    devuelve la longitud de la cadena

```

FIN **CCadena**

INTERFAZ CLASE **CLista**

TIPOS

TBase ...

METODOS

```

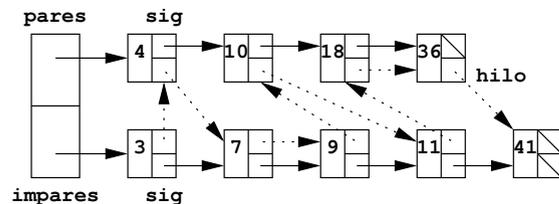
Crear()
Destruir()
Insertar(E N pos, E TBase x)
Eliminar(E N pos)
TBase Consultar(E N pos)
N Longitud()

```

FIN **CLista**

- 2 4pt La siguiente figura muestra una estructura pa-

ra mantener una secuencia ordenada de números naturales.



La estructura está formada por dos listas ordenadas, una para los pares y otra para los impares. Las listas de pares e impares están simplemente enlazadas a través de un enlace **sig**. Además, los nodos de la estructura contienen un segundo campo de enlace **hilo**, que permite recorrer los elementos del conjunto de forma ordenada, independientemente de su paridad.

Implementa un algoritmo **Inserta(E N x, ES TLista l)** que inserte un natural  $x$  en la lista  $l$ . Divide tu algoritmo en dos partes bien diferenciadas:

- la primera parte inserta ordenadamente  $x$  en la lista que le corresponda, según su paridad
- la segunda parte enlaza el nuevo nodo que contiene  $x$  en la lista mantenida por el campo de enlace **hilo**

Ten en cuenta que la estructura puede contener elementos repetidos.

- 3 3pt Codificar una función que tenga un árbol binario como parámetro de entrada y devuelva el número de elementos que hay en el nivel con más elementos.