



UNIVERSIDAD DE MÁLAGA  
E.T.S.I. TELECOMUNICACIÓN

Programación modular  
18 de junio de 2004  
(1º de Ingeniería de Telecomunicación)

Alumno:

Grupo:

(1.5 pts.)

- Dibujar los 14 árboles cuyo recorrido en inorden coincide con la lista  $[A, B, C, D]$ .
  - Dibujar el árbol cuyos recorridos en inorden y postorden son, respectivamente,  $[F, D, C, E, G, J, A, I, H, B, M, K, Z]$  y  $[F, D, E, J, G, C, I, H, B, A, Z, K, M]$ .
- Hacer una implementación no acotada del TAD CCola: (3 pts.)

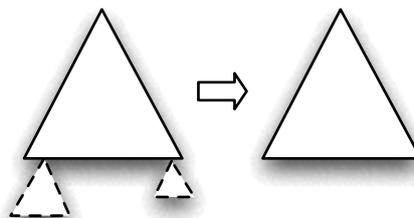
```
INTERFAZ CLASE CCola
TIPOS
  TBaseC ...
CONSTANTES
  N ...
METODOS
  Crear()
  Destruir()
  Encolar(E TBaseC x)
  TBaseC Desencolar()
  B EstaVacía()
FIN CCola
```

pero aprovechando más cada nodo. Para ello en cada nodo se podrán tener hasta un máximo de  $N$  datos  $TBaseC$ . Nótese que sólo los nodos extremos donde están los datos a desencolar y en el que se encola (principio y fin de la cola) pueden estar parcialmente ocupados; el resto de los nodos tendría las  $N$  celdillas ocupadas.

- Codificar en pseudolenguaje el algoritmo: (2.5 pts.)

ALGORITMO LlenoMásProfundo(EA ArbolBin a)

Este algoritmo poda las hojas o ramas que sean necesarios de los últimos niveles del árbol que se le pasa como parámetro para conseguir dejar el árbol **lleno** (igualado) más alto que se pueda partiendo del árbol original.



NOTA: Como definición de árbol binario puede usarse directamente, sin interfaz:

TIPOS

```
REGISTRO Nodo
  TElem dato
  Nodo *izq, *der
FIN
Nodo *ArbolBin
```

4. Dado el tipo `CListaPos`, construir el TAD `CPolinomio` cuya interfaz se expone a su lado sobre el TAD `CListaPos`. Esto es, los monomios del polinomio estarán contenidos en una lista dentro de `CListaPos`: (3 pts.)

INTERFAZ CLASE `CPolinomio`

TIPOS

`TBaseL ...`

METODOS

`Crear()`

`Destruir()`

`Copiar(CPolinomio orig)`

`AgregarMonomio(E N grado, E R coef)`

`TBaseP GradoPolinomio()`

`R Coeficiente(N grado)`

`Sumar(E CPolinomio sum, S CPolinomio res)`

`TBaseP Evaluar (E TBaseL valor)`

FIN `CPolinomio`

INTERFAZ CLASE `CListaPos`

TIPOS

`TBaseL ...`

METODOS

`Crear()`

`Destruir()`

`Insertar(E N pos, E TBASE x)`

`Eliminar(E N pos)`

`TBaseL Consultar(E N pos)`

`N Longitud()`

FIN `CListaPos`

El método `Copiar` copia el polinomio que se pasa como parámetro en el polinomio sobre el que se ejecuta el método. El método `AgregarMonomio` añade un nuevo término al polinomio. Si ya había un coeficiente con el mismo grado se le suma. En el polinomio sólo se almacenan aquellos grados cuyo coeficiente es distinto de 0. El método `GradoPolinomio` devuelve el grado del polinomio, es decir, la mayor potencia de la variable cuyo coeficiente es distinto de 0. El método `Coeficiente` devuelve el coeficiente del grado que se pasa como parámetro en el polinomio. Si no hay ningún valor para ese grado, se devuelve 0. En el método `Sumar` el segundo parámetro debe acabar almacenando la suma del polinomio sobre el que se aplica el método y el primer parámetro. El método `Evaluar` devuelve el valor de evaluar el polinomio en el punto que se pasa como parámetro

Observa que eres implementador de la clase Cola de prioridad y usuario de la clase Lista posicional.