

## Práctica 2: Árboles genéricos

**Ejercicio 1:** Aumentar la funcionalidad de la clase `LinkedTree` implementando los siguientes métodos:

```
/**
 * Returns an iterable collection of the the leaf nodes
 */
public Iterable<Position<E>> front();

/**
 * Returns an integer with the depth of the tree
 */
public int depth();

/**
 * Returns an integer with the degree of the tree
 */
public int degree();
```

Además, se deberán definir pruebas unitarias asociadas a estos tres métodos, definiendo al menos un caso de test JUnit por cada nuevo método.

**Ejercicio 2:** Implementar todos los métodos de las clases `LCRSTree` y `LCRSTreeNode` (implementación basada en hijo-hermano) como estrategia alternativa de diseño de árboles genéricos.

**Ejercicio 3:** Utilizar los test unitarios definidos para `LinkedTree` sobre `LCRSTree` y comprobar su correcto funcionamiento.

**Ejercicio 4:** Se desea desarrollar un programa en Java que permita analizar el contenido de un directorio y, de una manera interactiva, el de cualquier subdirectorio contenido en el directorio que se esté analizando en cada momento.

El siguiente fragmento de código muestra un ejemplo de interacción con el programa que se pide construir:

```
c:/User/jvelez/Documents/EDA/Practica 1> java -jar DirAnalyzer.jar
```

```
Introduzca el nombre del directorio a analizar: c:\ProgramFiles
Analizado 100%
```

```
c:\ProgramFiles - 589 MB
2 - Office 2013 - 320 MB - 54%
3 - Gimp - 54 MB - 9%
4 - Corel Draw - 215 MB - 36%
```

```
Introduzca el número del subdirectorio a explorar o 0 para salir: 2
Office 2013 - 320 MB
1 - ..
2 - Word 2013 - 120 MB - 36%
3 - Excel 2013 - 105 MB - 32%
4 - PowerPoint 2013 - 105 MB - 32%
```

```
Introduzca el número del subdirectorio a explorar o 0 para salir: 1
c:\ProgramFiles - 589 MB
2 - Office 2013 - 320 MB - 54%
3 - Gimp - 54 MB - 9%
4 - Corel Draw - 215 MB - 36%
```

```
Introduzca el número del subdirectorio a explorar o 0 para salir: 0
Que tenga un buen día.
c:/>
```

Con objeto de que la etapa interactiva sea ágil, se desea que el programa explore y almacene en un análisis inicial todo el contenido del directorio que se le indique.

Para almacenar el resultado de la exploración se recomienda que el programa genere un árbol. Los nodos hoja de dicho árbol corresponderán a los ficheros, y almacenarán el nombre del directorio y el tamaño en bytes del archivo que representan. Por otro lado, cada subdirectorio intermedio se corresponderá con un nodo interno, que almacenará el nombre del subdirectorio y la suma de los tamaños de todos los nodos hojas descendientes de él.

Así, en la etapa interactiva, el programa se dedicará a recorrer el árbol según las peticiones del usuario.