



**Universidad de los Andes**  
Ingeniería de Sistemas y Computación  
ISIS 1205 - Algorítmica y Programación 2  
Taller teórico nivel 11



**Problema 1 [40%]** En la clase **Ciudad** adicionar el método **recursivo** que por cada ciudad convierte su subárbol 3-ario en árbol binario de acuerdo a los siguientes casos:

**Caso General:** Si su ciudadCentral no hace parte del camino respuesta de la solución al juego, la ciudad modificara sus ciudades hijas para quedar únicamente con 2 ciudades ordenadas:

- La ciudadCentral se elimina (valor null) incluyendo todo el subárbol que contenga.
- La ciudad menor entre la ciudadIzquierda y la ciudadDerecha quedará en la ciudadIzquierda.
- La ciudad mayor entre la ciudadIzquierda y la ciudadDerecha quedará en la ciudadDerecha.
- El criterio de orden para comparar 2 ciudades es el nombre de la ciudad.

**Caso Especial:** Si su ciudadCentral hace parte del camino respuesta de la solución al juego, se debe eliminar la ciudadIzquierda o la ciudadDerecha:

- De estas 2 ciudades la ciudad eliminada será aquella que tenga el menor número de lugares donde hay pistas. Si ambas ciudades tienen el mismo número de pistas, la ciudad eliminada será la ciudadIzquierda.
- De estas 2 ciudades la que queda junto con la ciudadCentral deben garantizar el orden como hermanas: la ciudad menor quedará en la ciudadIzquierda y la ciudad mayor en la ciudadDerecha.

*Ayuda:* Utilizar el método **enCaminoRespuesta()** de la **clase Ciudad** que retorna true si la ciudad hace parte del camino respuesta para encontrar al sospechoso y false en caso contrario.

**Problema 2 [30%]** En la clase **Ciudad** agregar el método **recursivo** que retorna el tiempo de viaje (en días) que le toma a la Interpol recorrer su rama más demorada sin utilizar pistas. En este ejercicio considere que cada ciudad es un árbol 3-ario.

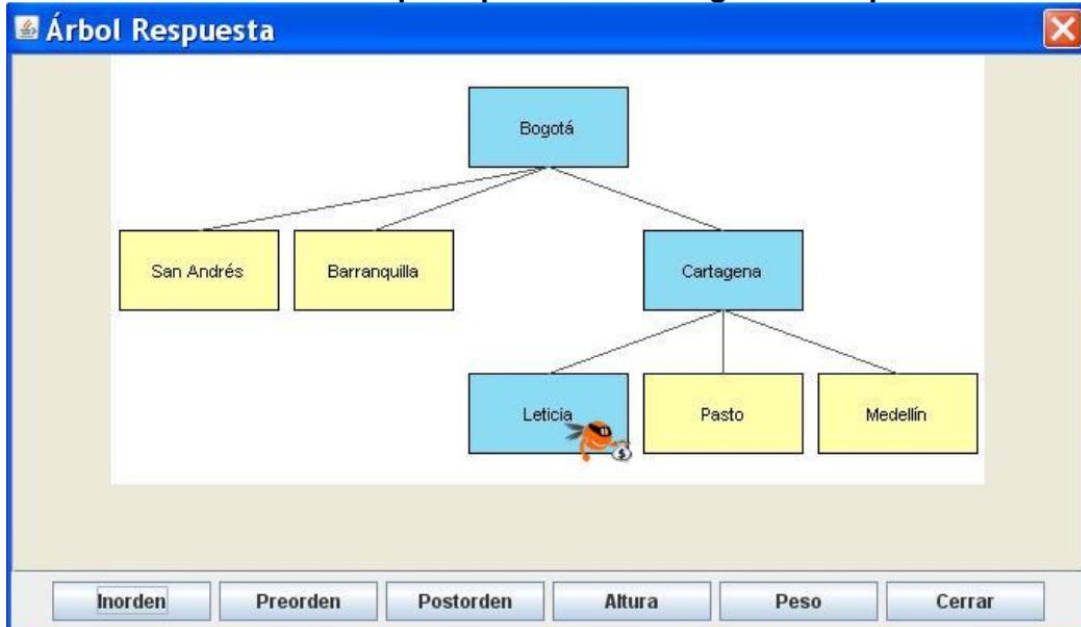
*Ayuda:* Utilizar el método **darTiempoViaje()** de la **clase Ciudad** que retorna el número de días (entero) de viaje desde la ciudad padre hasta la ciudad que llamó al método.

**Problema 3 [30%]** En la clase **Ciudad** agregar el método **recursivo** que define la lista de nombres de ciudades (Strings) hojas donde se puede perder el juego. En cualquier ciudad hoja donde no se encuentre el sospecho se pierde el juego. El método usa un parámetro de la interfaz **List** para acumular las ciudades que cumplen la propiedad. En este ejercicio considere que cada ciudad es un árbol 3-ario.

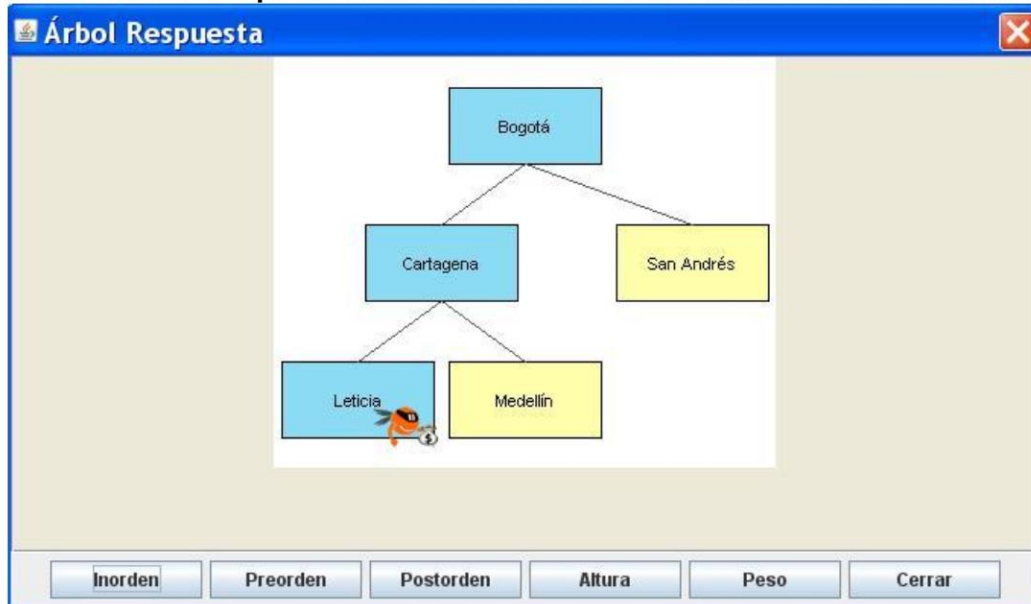
*Ayuda1:* Utilizar el método **esHoja()** de la **clase Ciudad** que retorna true si la ciudad es una hoja del árbol y false en caso contrario.

*Ayuda2:* Utilizar el método **estaSospechoso()** de la **clase Ciudad** que retorna true si en la ciudad se encuentra el sospechoso buscado y false en caso contrario.

Ilustración del Arbol 3-ario que representa el Juego de Interpol versión fácil.



Ejercicio 1. Solución esperada a la conversión a Arbol Binario Parcialmente Ordenado



Ejercicio 2. Solución esperada a la duración en días de la rama más demorada del árbol 3-ario.

Caso juego fácil: 4 días  
 Caso juego intermedio: 4 días

Ejercicio 3. Solución esperada a la lista de ciudades hojas donde se pierde el Juego en árbol 3-ario

Caso juego fácil: "San Andrés", "Barranquilla", "Pasto", "Medellín"