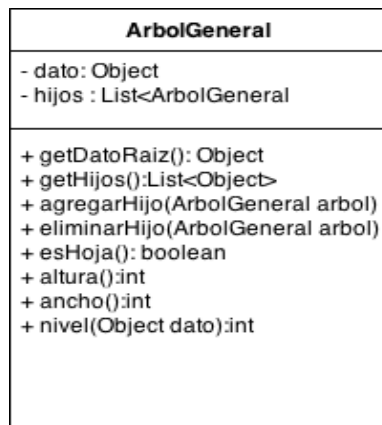


## Práctica 3: Estructuras de Datos y Algoritmos II UPE 2015



1- Implemente en la clase ArbolGeneral los siguientes métodos:

**a) eliminarHijo(ArbolGeneral arbol)** quien recibe el mensaje elimina el hijo pasado por parámetro

**b) esHoja(): boolean** devuelve true si el nodo en cuestión no tiene hijos, false en caso contrario.

**c) altura(): int** devuelve la altura del árbol, es decir, la longitud del camino más largo desde el nodo raíz hasta una hoja. Pista: el mensaje altura debe chequear si el árbol es una sola hoja o no. Si el árbol es una sola hoja, se devuelve 0. Si no, se utiliza el mensaje getHijos() para obtener la lista de hijos (recuerde que devuelve una lista de árboles hijos). Luego, debe iterar por cada uno de los hijos, guardando la máxima altura. A este valor se le debe sumar 1 y retornarlo.

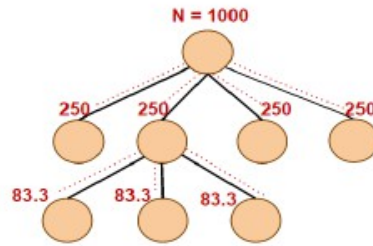
**d) nivel(Object dato): int** devuelve la profundidad o nivel del dato en el árbol. El nivel de un nodo es la longitud del único camino de la raíz al nodo. Pista: si el nodo raíz posee el mismo dato que pasado como parámetro, se retorna 0. En caso contrario, se debe buscar en cuales de los subárboles hijos se encuentra el dato (implemente el mensaje **include (Object dato)** en la clase Arbol General) y se debe retornar 1 más el nivel que arroje enviar el mensaje nivel() al subárbol que incluye el dato.

**e) ancho(): int** la amplitud (ancho) de un árbol se define como la cantidad de nodos que se encuentran en el nivel que posee la mayor cantidad de nodos. Pista: realice un recorrido por niveles. Encole inicialmente la raíz del árbol y luego una marca None (o el número de nivel) para indicar el fin de nivel.

Mientras la cola no se vacía, itere. En cada iteración extraiga el tope de la cola, y con la operación getHijos() encole los mismos. Cuando encuentra la marca de fin de nivel cuente si los elementos del nivel es mayor a la máxima cantidad que poseía.

**Los siguientes ejercicios son aplicaciones que “usan” el árbol general.**

2- Sea una red de agua potable, la cual comienza en un caño maestro y el mismo se va dividiendo sucesivamente hasta llegar a cada una de las casas. Por ejemplo, si en el caño maestro ingresan 1000 litros, a medida que el caño se divide, el caudal se divide en partes iguales en cada una de las divisiones. Es decir, si el caño maestro se divide en 4 partes, cada división tiene un caudal de 250 litros. Luego, si una de esas divisiones se vuelve a dividir en 3 partes, cada una tendrá un caudal de 83,3. La situación descrita se puede modelar de la siguiente forma a través del siguiente árbol general:



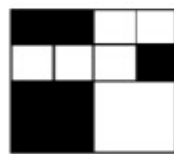
Usted debe implementar la clase `RedDeAguaPotable`, la cual está compuesta por una estructura de árbol general, y además contiene un método llamado **minimoCaudal**, que recibe un valor `N`, correspondiente a los `N` litros que ingresan por el caño maestro, y devuelve cuál es el mínimo caudal que recibe una hoja.

3- Un quadtree es una representación usada para cubrir un espacio cuadrado en dos dimensiones y posteriormente utilizado para determinar ciertas condiciones entre objetos en el mismo.

Un artista moderno trabaja con imágenes codificadas en quadtrees. El quadtree es un árbol 4-ario que codifica a una imagen con el siguiente criterio:

- Si toda la imagen tiene un mismo color, la misma es representada por un único nodo que almacene un dato que represente a ese color.
- En caso contrario, se divide la imagen en cuatro cuadrantes que se representan en el árbol como un nodo con 4 hijos, y cada hijo es la conversión de cada una de las partes de la imagen.

El artista desea saber cuántos pixeles de color negro posee una imagen dada. Usted debe implementar un método, que dado un quadtree y una cantidad total de pixeles, cuente cuantos pixeles de color negro contiene la imagen codificada en él.



Para el quadtree de la Figura, la salida del método sería 448

La figura muestra un ejemplo del árbol `quadTree` correspondiente a la imagen de la izquierda de 32 x 32 pixeles (1024 pixeles en total). Cada nodo en un quadtree es una hoja o tiene 4 hijos. Los nodos se recorren en sentido contrario a las agujas del reloj, desde la esquina superior derecha.

4- Jack es australiano y está de visita por la Argentina. Como todo turista quiere aprovechar al máximo su estadía. Su vuelo arriba a Buenos Aires y debe decidir que lugares visitar. Lo que sí sabe es que tomará una de las rutas nacionales que parten de Bs.As. y se alejará de la misma hasta llegar a la frontera con algún de los países limítrofes.

Considerando que los caminos posibles tienen forma de árbol en donde cada tramo de ruta tiene una cierta longitud y las ciudades tienen un número de sitios turísticos. Usted debe implementar un algoritmo para que determine qué camino debe recorrer, sabiendo que parte desde la raíz del árbol (ciudad de Buenos Aires) forma de minimizar la distancia recorrida, y si dos caminos tuvieran la misma distancia, deben maximizar los sitios turísticos (suma total de todos los sitios turísticos de las ciudades que atraviesa el camino)

Implemente la clase `Lugar` donde tendrá el atributo `cantidadDeSitios`, la clase `País` donde tendrá el atributo `rutas` que será un `ArbolGeneral` de objetos `ciudad`, y un método en esta última para buscar el mejor recorrido de acuerdo a lo planteado en el enunciado.